AD-A049 225 E-SYSTEMS INC ST PETERSBURG FL ECI DIV F/G 17/2.1

TRANSMISSION SUBSYSTEM CONTROL ANALYSIS AND DEVELOPMENT.(U)

JUL 77 R K SMITH, J N BEAUCHAMP DCA100-76-C-0036

UNCLASSIFIED SBIE-AD-E100 012 NL

1 OF 3

AD
A049225

SIFIED

1 OF 3

AD
A049225

1.0 2.8 2.5 3.2 2.2 3.6 2.0 4.0 1.1 1.8 1.25 1.4 1.6

MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

AD-E 100 012

FINAL REPORT

TRANSMISSION SUBSYSTEM CONTROL

E-SYSTEMS
ECI Division

ST. PETERSBURG, FLORIDA

9 FINAL REP[redacted]RT. 23 Feb 76 - 15 Jul 77

6 TRANSMISSION SUBSYSTEM CONTROL
ANALYSIS AND DEVELOPMENT.

10 Richard K. Smith
Jere N. Beauchamp

PREPARED FOR

DEFENSE COMMUNICATIONS AGENCY

CONTRACT NO.

15 DCA 100-76-C-0036

11 15 JUL[redacted]77 12 274 p.

18 SBIE 19 AD-E100 012

PREPARED BY

E-SYSTEMS
ECI Division

ECI DIVISION OF E-SYSTEMS, INC.
P. O. BOX 12248
ST. PETERSBURG, FLORIDA 33733 U.S.A.

mt

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) TRANSMISSION SUBSYSTEM CONTROL ANALYSIS AND DEVELOPMENT		5. TYPE OF REPORT & PERIOD COVERED FINAL REPORT 2/23/76 to 7/15/77
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) RICHARD K. SMITH JERE N. BEAUCHAMP		8. CONTRACT OR GRANT NUMBER(s) DCA 100-76-C-0036 new
9. PERFORMING ORGANIZATION NAME AND ADDRESS E-Systems, ECI Division 1501 72nd Street North St. Petersburg, FL 33733		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Defense Communications Agency 8th and South Courthouse Road Arlington, VA 2220.		12. REPORT DATE 15 July 1976
		13. NUMBER OF PAGES 264
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office) Defense Communications Engineering Center 1860 Wiehl Avenue Reston, VA 22090		15. SECURITY CLASS. (of this report) UNCLASSIFIED
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Transmission Control
Fault Isolation
Data Communications
Data Acquisition

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

This report represents the results of a study aimed at analyzing the needs and developing a concept for transmission control for the DCS Digital European Backbone (DEB) Network. Included in this report are considerations regarding operations concept, telemetry, data acquisition, processing, control and reporting.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

TABLE OF CONTENTS

TABLE OF CONTENTS (Continued)

TABLE OF CONTENTS (Continued)

LIST OF ILLUSTRATIONS

LIST OF ILLUSTRATIONS (Continued)

LIST OF ILLUSTRATIONS (Continued)

LIST OF TABLES

PRECEDING PAGE BLANK - NOT FILMED

1.0 INTRODUCTION

This report summarizes the results of a study of the requirements for Transmission Subsystem Control for the DCS Digital European Backbone (DEB) network. As a model for this study, the Projected European DCS Connectivity 1982 was used.

The most important requirement of the TSC is the remoting of alarms, monitors and control points associated with equipments at remote sites to manned locations where actions can be taken. A secondary goal of the TSC is the improvement of circuit availability by the implementation of automatic fault isolation and restoral algorithms.

Included in this report are considerations regarding operations concept, telemetry, data acquisition, processing, control and reporting.

2.0 BACKGROUND AND SUMMARY OF PREVIOUS RESULTS

The initial tasks for this contract consisted of an analysis and characterization of the overall availability of the DCS digital transmission subsystem. A transmission subsystem control algorithm was to be developed based on the results of the availabiltiy analysis. This algorithm was to use alarms and monitors available from the radio, multiplexer and cryptographic equipment to isolate faults and to initiate appropriate fault correction. An important consideration in the development of this algorithm was the control of the bypass which is built into the Walburn cryptographic equipment. This bypass when activated causes a failed cryptographic equipment to by bypassed. However, initiation of the bypass requires an external command and it is the external control of the bypass function which was investigated.

Three scenarios, representing three levels of control sophistication, were considered. These are defined as follows:

- Scenario A - The TSC algorithm can make use of measurements of all local site equipment alarms and monitors but exercises control of Walburn resync and Walburn bypass only.

- Scenario B - The TSC algorithm can make use of measurements of all local site equipment alarms and monitors and can perform redundant swtiching and control of all local equipment.

- Scenario C - In addition to monitoring and controlling local equipment, the TSC uses the service channel to gather alarm data from other sites and to command remote switching and control.

The results of a study of Scenarios A and B were reported in the ECI Design Plan dated 23 September 1977. These results showed that the availability gains that could be realized with these levels of control were not significant.

One approach of KG-81 bypass and resync control was presented in Section 6 of the Design Plan. With regard to control of the KG-81 bypass, it was concluded that bypass action should be initiated immediately if certain unequivocal alarms such as Walburn Primary Power or Summary Alarm occur. In cases where the syndrome does not unequivocally indicate a Walburn failure, it is appropriate to relegate bypass action to the last action of the restoral action list. Penalty-free restoral attempts should be taken first since bypassing the Walburn will cause a temporary bi-directional outage and results in clear-mode operation.

The condition for commanding a KG-81 resync action is the existance of an MSB outage with the condition that the FRC-163 is in sync and the associated TD-1193 is frame alarming. This same syndrome could, however, be caused by radio port hardware, a TD-1193 failure or a KG-81 failure. Since KG-81 resync action introduces a two-way outage, it was concluded that penalty-free switching actions should be attempted first. A flow chart for these control actions is given in Section 6 of the Design Plan.

Without remote data acquisition and control, circuit availability is so dominated by the unavailability contributions of the Level 1 Multiplexers and unmanned radios that little can be gained by automatic control of the Walburn bypass. The preliminary conclusion was that the level of transmission control considered in Scenarios A and B was not justified. Accordingly, the contract effort was redirected to provide for a greater emphasis on the study of Scenario C in lieu of brassboard model development.

The telemetry capability assumed in Scenario C is essential to the control of the Walburn bypass at remote, unmanned sites whether the control is manually or automatically initiated. With remote control of the bypass, the availability of the Walburn/Walburn bypass combination in unmanned configurations can be made to approach the availability of this equipment in manned configurations.

Scenario C expands the scope of the problem to encompass those aspects of Automated Technical Control (ATEC) that are generally known as "digital ATEC." In other words, the redirected study involves a systems look at the general monitoring, control and fault isolation problem for DEB. Transmission control that performs these functions appears to be required as an aid to manual fault isolation regardless of what improvement in availability is afforded by automatic fault isolation.

The expanded scope of Scenario C impacts nearly all aspects of the study. One important new consideration is the development of a concept for utilizing the 56 Kbps digital service channel to support the TSC functions. Secondly, an automatic fault isolation algorithm that has access to remote stream-related alarm and monitor data and can troubleshoot faults by performing remote switching actions has a significiantly higher lever of sophistication than an algorithm restricted to local control. The hardware and software needed for processing, data acquisition and telemetry control is also impacted.

This Final Report presents the results of the investigation of the Scenario C TSC concept.

3.0 SYNOPSIS

This report presents the results of a study aimed at analyzing the needs and developing a concept for a transmission subsystem control (TSC) for the DEB network. The highlights of the report content include:

- A concept for a low-cost transmission subsystem control that makes use of distributed processing. The system can function autonomously at the nodal level or be placed under the discipline of a control hierarchy.
- A concept to use the service channel as a message switched subnetwork to support transmission control and other system services.
- Data acquisition requirements to support transmission control.
- An automatic fault isolation algorithm for isolating unalarmed faults.
- A design plan for implementing the TSC concept.
- Estimated cost of the proposed system.

One of the major advantages of the low-level distributed control concept presented here is that the effectiveness of this control can be evaluated by the deployment of hardware over a small segment of the network. It doesn't require the committment of funds to a multi-million dollar program at the

outset as would a highly centralized transmission control concept. The system presented here can be deployed in stages.

Another significant feature is the flexibility of the TSC architecture. With the exception of the off-loading of the communications protocol functions to a powerful LSI device, virtually anything that anyone would ever want to alter is in software. The TCU, for example, is a microcomputer with two very general and one specialized perpherial. These perpherials are respectively, a Data Acquisition Subsystem, a Control/Display Unit (CDU) and the Telemetry Subsystem.

The Data Acquisition Subsystem provides rapid scanning and alarm change detection in hardware. However, it is also general purpose in that any data or control point can be addressed under software control.

The CDU is a general purpose intelligent terminal with a standard serial interface. It is envisioned that alterable firmware internal to the CDU will contain message formatting, operator lead-through and entry validation routines.

The recommended Telemetry Subsystem makes use of a highly structured hierarchy of communication protocols. Separation of functions (link, network and user related) results in a very flexible communications capability.

The telemetry subsystem concept developed here is, in fact, a general purpose, message-switched network capable of establishing a digital message exchange capability between any pair of stations in the DEB network. The telemetry

organization features the adaptive allocation of service channel resources under software control. Although the TSC mission exploits the available 56 Kbps capacity to full advantage when there are no outside demands, the actual communications resource needs of the TSC are small leaving ample service channel resource for other missions. This versatile communications subnetwork could be used, for example, to provide:

- A DEB-wide TTY network
- Control for circuit testing
- Remote patching for rerouting priority digroups

4.0 CONCLUSIONS

The results of this study show that the TSC can be an invaluable aid in increasing maintenance effectiveness and reducing tech controller workload, required skill level and manpower requirements. Its main value lies in the fact that it gives tech controllers accessability to remote (unmanned) site alarm and monitor data and control points. For example, the DRAMA equipment has a built-in fault detection and redundant unit switchover capability. When such a switchover occurs at a remote site, however, this fact will not be known without the remote data acquisition capability provided by the TSC. It is important that the responsible tech controller be informed of the switchover so that the failed standby unit can be repaired. Another case where the TSC is of great value is when a failure occurs somewhere in a chain of unmanned repeaters. Without a remote data acquisition capability, it may be impossible to isolate the failure to a single site in the chain in order to dispatch a maintenance team to the specific site where the problem exists.

The TSC can provide other valuable assistance in the task of fault diagnosis by informing the tech controller of the unsuccessful equipment switching actions that have been automatically taken and suggesting to the controller a list of remaining possible causes.

The circuit availability gains that can be achieved by an automatic fault isolation and restoral algorithm are limited by 1) the availability of the TD-1192 multiplexer and 2) the non-redundant components of the TD-1193 and the FRC-163. (The automatic control action of the TSC in restoring an equipment failure is limited to redundant unit switching. There is no automatic action that can be taken by the TSC to restore an

outage due to a non-redundant equipment failure.) If realistic assumptions are made regarding the degree of non-redundant circuitry in the TD-1193 and FRC-163, the availability gains appear to be insufficient to warrant the deployment of a TSC for this reason alone.

An important result of this study is that it demonstrates that the state-of-the-art in computer-to-computer communications is such that a flexible, high performance message-switched subnetwork can be economically configured to operate over the 56 Kbps digital service channel. This telemetry subnetwork can effectively support the data acquisition and remote control needs of the TSC. At the same time, it affords a surplus communications capacity that can be utilized for other system functions. The telemetry subsystem design presented makes use of state-of-the-art, multi-protocol receiver/transmitter LSI devices to shift a substantial communications overhead processing load from software to low-cost hardware.

Analytical results are presented which show that, even during a failure episode, the telemetry subsystem loading is a small fraction of the total capacity.

The loading of the TSC processing function is also quite low. It is demonstrated that an economical, single-processor design is possible for both the Transmission Control Unit (to be deployed at nodes) and the Remote Telemetry Unit (to be deployed at repeaters).

The hardware and software implementations have been studied in sufficient detail to provide the firm conclusion that the development risk is low. All components required for the suggested implementation are readily available.

Finally, it is concluded that a distributed monitoring and control system that functions autonomously at the nodal level is a viable solution to the transmission control problem. This type of control system can be deployed in a staged fashion and permits full-up, field evaluation of the concept without a large hardware procurement. A moderate amount of non-recurring hardware and software engineering is required but the development risk and hardware production costs are relatively low.

5.0 OPERATIONS CONCEPT

This section gives an overview of the transmission subsystem control concept that has resulted from this study. Many of the aspects of the subsystem that are touched upon in this section are treated in greater depth in succeeding sections of the report.

5.1 TSC Required/Desired Capabilities

The main purpose of the TSC is to provide survivable monitoring and control of the transmission equipment. A secondary purpose is to provide some degree of processing of alarm and monitor data, performing automatic fault isolation or aiding in manual fault isolation with an aim toward improving circuit availability. Specific capabilities that are required to carry out this mission include:

- Relatively high-speed scanning of transmission equipment alarms that are critical fault indicators.
- Reporting to manned locations, critical alarm and status changes that occur at remote sites.
- Remote access to transmission equipment control points.
- Remote access to all transmission equipment alarm and monitor data that may aid in fault isolation.
- Automatic processing of alarm and status change data in order to isolate failures.
- Initiation of automatic control actions aimed at service restoral when the failure results in a service outage.
- Reporting results of diagnosis and restoral attempts to the tech controller.

5.2 Control Concept

Previous work related to transmission control has considered varying degrees of hierarchial control. One point of agreement has been that any hierarchial control system should be capable of a fall-back mode where lower level controllers operate (perhaps sub-optimally) in an autonomous fashion.

It is also generally recognized that it is desirable, from the standpoint of survivability and telemetry channel loading, to distribute as much of the processing and control as is feasible to the lowest practical level.

With regard to hierarchial control considerations, it is observed that many of the factors that influence a decision regarding an appropriate control hierarchy are related to system management, maintenance doctrine, etc. A decision cannot be made based purely on technical and economic considerations.

Secondly, it is recognized that a number of functions required in a transmission control subsystem (representing a substantial part of the system engineering task) are more or less independent of any hierarchial considerations. These include the data acquisition function and the telemetry function.*

With microprocessor technology, it turns out that the sensible place for the processing and control needed for fault isolation is at the nodal level independent of hierarchial considerations. (Fault isolation is basically circuit

*NOTE: It is recognized that different control concepts will impact telemetry channel loading. In the DEB, however, the telemetry channel resource (56 kbps full duplex) is more than adequate to accommodate centralized processing and control.

oriented and the algorithms rely heavily on the correlation of digroup alarms. The points of confluence for digroups - and mission bit streams - are at the network nodal points.)

It has thus been possible to engineer a self-sufficient transmission subsystem control largely independent of hierarchial considerations but which, at the same time, can logically be subjected to the discipline of a control hierarchy.

(In terms of the control hierarchy outlined in DCEC TR-5-74, the present study addresses the implementation of a full capability transmission control subsystem that is capable of autonomous operation at the third level of the hierarchy.)

5.3 DEB Network Characteristics

The general characteristics of the DEB network that forms the basis for this study are presented in this section.

5.3.1 General Attributes

The DCS Digital European Backbone (DEB) network is being deployed in stages. The baseline for this study is the Projected European DCS Connectivity - 1982.

The first important attribute of the DEB is due in part to this staged deployment. This attribute is the fact that the general structure and specific connectivity is subject to change. This implies that it is imperative that the TSC easily accommodate change.

The model network - as it will exist in 1982 - is comprised of 115 stations, 30 of which are simple, unmanned repeater sites. The other 85 stations span a wide range of station complexity varying from relatively simple drop and insert repeaters to major network nodes such as Donnersburg which terminates 13 branches.

The DEB is not a switched network but is comprised of point-to-point dedicated digroups. Widely separated digroups are often multiplexed into common mission bit streams and link streams. It is also noted that there are instances where two digroups having the same end-points traverse distinct network paths. Although there is no automatic switching, digroup connectivity can be reconfigured by manual patching.

With regard to network topology, these are often alternate independent radio paths between two stations, but this is by no means generally true.

Radio links between stations are usually microwave but can be troposcatter. Radios can be configured either in a frequency diversity, space diversity or hot standby configuration.

5.3.2 Station Types

The model used for this study was the Projected European DCS Connectivity - 1982 and associated Stage I through Stage IV Mux Plans. For the purpose of this development, six distinct station types are recognized. These are defined as follows:

STATION TYPE	DESCRIPTION
S	Main station with system responsibility
A	Main station with regional responsibility
B	Main station: subordinate
C	Branching repeater (unmanned - no vf drops)
D	Drop & insert repeater
E	Baseband repeater (unmanned)

A representative segment of the network is shown in Figure 5-1. The network can be viewed as a collection of relatively main stations that are interconnected either directly or via chains of repeaters by radio links (line-of-site or tropo). Sites types A, B and S are manned stations having an appreciable amount of transmission equipment and which frequently terminate three or more branches. At most of the branches at these site types, there is a confluence of digroups forming a Mission Bit Stream (MBS) and, often, a confluence of two MBS's forming a "Link Stream."

Site type C is an unmanned branching repeater characterized by the fact that it terminates three or more branches. Generally, Type C sites terminate no digroups but frequently demultiplex and re-multiplex digroups to provide branching at the digroup level. Occasionally, MBS's are through-grouped at Type C sites. In general, Type C sites are characterized by points of confluence of digroups and MBS's.

Site Type D has been termed a "Drop-and-Insert Repeater" since there are many instances where these sites serve in what is through of as a repeater role. Frequently, however, Type D sties are at the end of a "spur" making "repeater" a misnomer. In either case, a Type D site is a relatively minor site in terms of its equipment complement but always terminates one or more digroups. Type D sties never terminate more than two branches. In all instances, Type D sites are presumed to be manned.

A Type E site is a simple, unmanned repeater having an equipment complement that consists of two radio sets and two service channel multiplexers.

The distinction between site Types A, B, and S lies in the fact that it is recognized that certain sties have, in some sense, system responsibility (Type S) and that other sites (Type A) have some sort of regional responsibility. Other

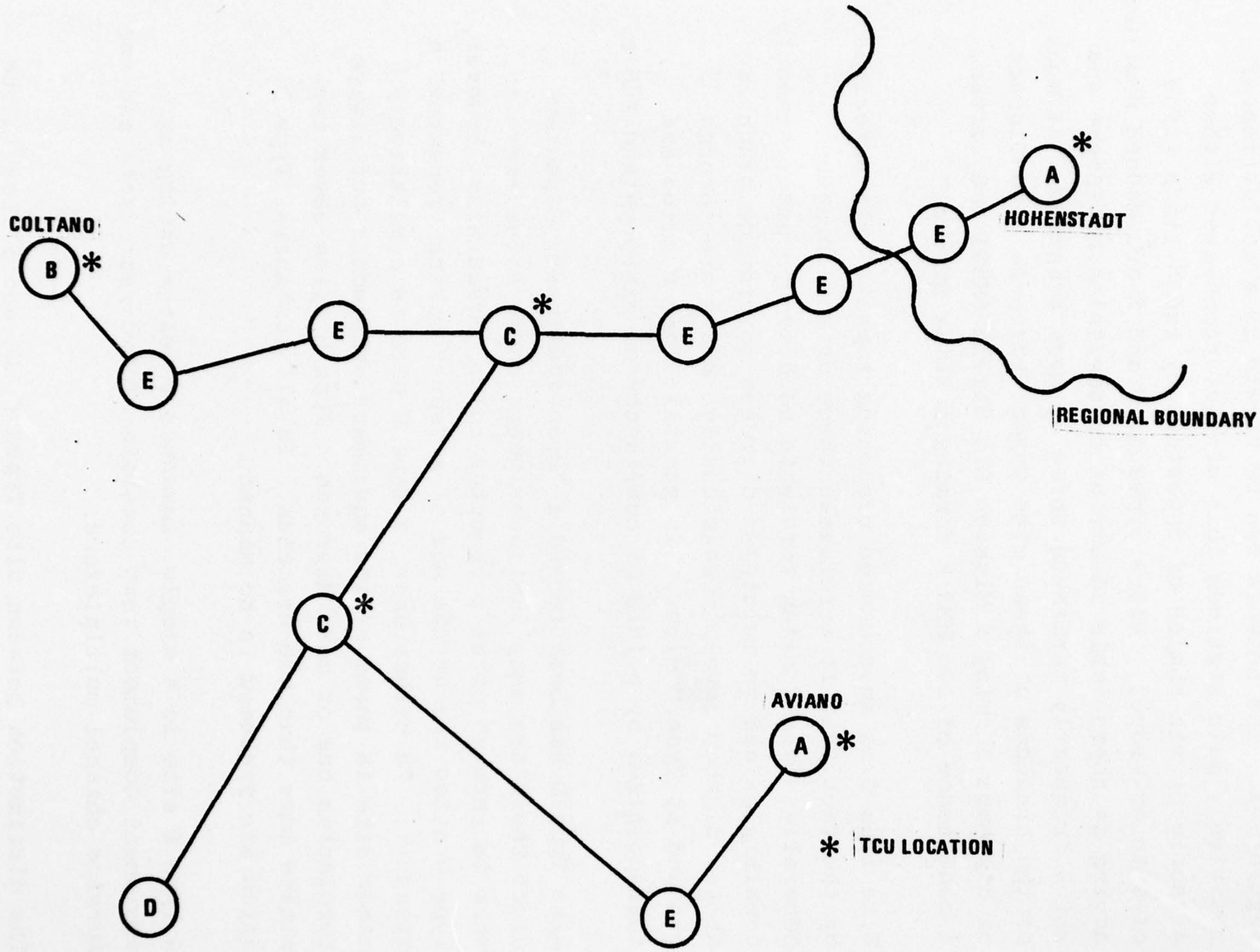


FIGURE 5-1
DEB NETWORK SEGMENT

main stations (Type B) have a subordinate role. (In configuring a transmission control subsystem that functions autonomously at the nodal level, no special significance has been attached to these system and regional sites with the exception that it has been assumed that some ancillary communications will flow between these and their subordinate stations. Such communications for example might be summary reports destined for a system or regional data base or higher level approvals of proposed control actions.)

5.3.3 Equipment Configurations

The station equipment, for the purpose of this study, has been assumed to be the normal DRAMA configuration with bulk encyrption applied at the MBS level. This standard configuration is illustrated in Figure 5-2.

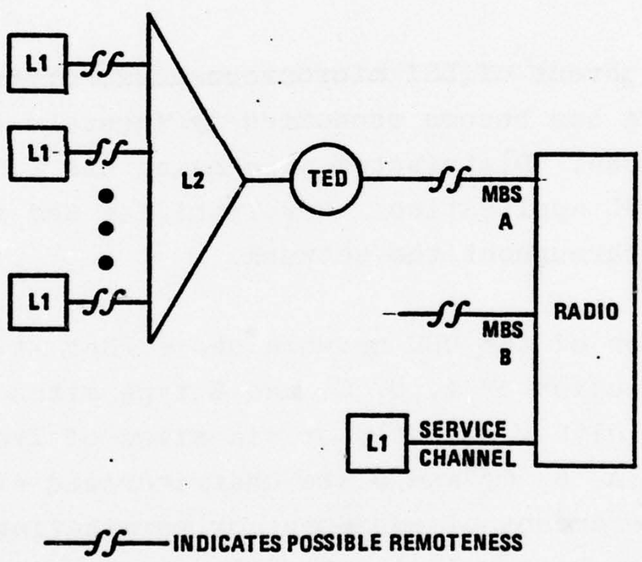


FIGURE 5-2
STANDARD EQUIPMENT CONFIGURATION

The TD-1193 and the FRC-163 are virtually fully redundant with switchover at the equipment level. The KG-81 is equipped with a clear-mode bypass. The TD-1192's are non-redundant. The service channel available to support transmission control is a full-duplex, 56 kbps digital channel connecting each adjacent station.

5.4 TSC Concept

The transmission control requirements outlined in Section 5.1 can be satisfied by the deployment of a TSC that is at the same time both low cost and highly survivable. As with any control system, the TSC includes the functions of data acquisition, processing and control. Data acquisition and control activation is, of course, needed wherever there is transmission equipment. Processing can be distributed or centralized.

With the advent of LSI microprocessors, distributed processing has become economically feasible in many control applications. Distributed processing has two main advantages for the TSC application: survivability and reduced flow of raw data throughout the network.

Examination of the DEB network shows that it can be viewed as a collection of A, B, C, and S type sites that are inter-connected either directly or via sites of Types D and E. Site Type A, B, C, and S are characterized either as having a sizeable amount of equipment or terminating three or more branches or both. It is the multiple branch feature that really sets these sites apart from site types D and E for transmission control purposes because it is mainly due to this characteristic that these sites are logical choices for the deployment of the processing function.

There are two primary reasons why a processing capability is needed at multi-branch sites. First, the reporting needs of the TSC imply a requirement for message switching at multi-branch nodes. The second reason is that it is at these multi-branch sites that digroups come together to form mission bit streams and MBS's come together to form link streams. It is at these points of confluence that a basis exists for performing alarm correlations.

Based on these considerations regarding the deployment of the processing function, a TSC concept has evolved in which a control subsystem is configured using two basic types of equipments: a Transmission Control Unit (TCU) deployed at site types A, B, C, and S and a Remote Telemetry Unit (RTU) deployed at site Types D and E. The deployment of TCU's define what will be referred to as "local loops." A local loop is said to exist between two adjacent TCU's and is the basic entity for data acquisition and control purposes. A local loop has a TCU at either extremity and an arbitrary number of intermediate RTU's.* The TCU's monitor and exercise control over the associated RTU equipped sites. TCU's can function autonomously or can be placed under the control of a higher level controller. The following functions are <u>required</u> at RTU's and TCU's.

RTU (Remote Telemetry Unit)

- Data Acquisition
- Remote Control Activation

TCU (Transmission Control Unit)

- Data Acquisition
- Alarm Correlation
- Fault Isolation
- Automatic Control
- Manual Remote Control
- Status Reporting
- Message Routing

*An exception is the case of a "spur" where the loop has a single TCU.

In the normal operating mode, the TCU's at either end of local loops use the service channel to poll or "scan" the stations of the loop for any changes that occur in a predefined set of critical alarms associated with each equipment. Detection of an alarm change will produce an automatic action by the responsible TCU that will ultimately result in the generation of a report directed to the cognizant TCF. There are three kinds of report that can result from the detection of an alarm change: Status Change Report, Failure Report, or Summary Alarm. If the alarm change indicates a stream outage, an automatic fault isolation routine will be executed by the responsible TCU in an attempt to restore service. If this restoral attempt is successful, a Failure Report will be generated designating the source of the fault; if not, a Summary Alarm will be generated.

A simple status change can occur, for example, when a routine switchover to a standby unit is performed for maintenance purposes. Nothing has failed, but the status change will be reported, for example, in a format similar to the following:

```
STATUS CHANGE REPORT
        ACTION:   TDM DEMUX SWITCHOVER
        ID:       DONNERSBURG/BRANCH 1/MBS A
        CURRENT:  ON-LINE - UNIT NO. 1
```

A Failure Report can result from either a failure that is detected by the transmission equipment itself (but which does not produce an outage) or can be the result of the successful restoral of an outage by the automatic fault isolation algorithm. In either case, it serves to designate a known failed piece of equipment that has been switched off-line and is not causing an outage. Examples of two Failure Reports are given below. The first represents a case where the failure was detected by the TD-1193's built-in fault detection circuitry and the faulty unit was

automatically switched off-line. The second example report illustrates the case of an unalarmed failure that was detected and the outage restored by the TSC algorithm.

FAILURE REPORT

UNIT:	TDM MUX
ID:	FELDBURG/BRANCH 3/MBS A/UNIT No. 2
MODE:	SELF-DETECTED
ALARMS:	FAULT MBS OUTPUT

FAILURE REPORT

UNIT:	FRC-163 TRANSMITTER (PORT RELATED)
ID:	FELDBURG/BRANCH 2/UNIT No. 1/PORT B
MODE:	TSC RESTORAL
RESTORAL TIME:	820 MSEC
ALARMS:	NONE

Summary Alarms result when an outage occurs that the TCU is unsuccessful in restoring. After all appropriate isolation/restoral actions have been attempted, a report in a format similar to the following will be issued.

SUMMARY ALARM:

STREAM:	DONNERSBURG/BRANCH 2/MBS A/DIGP 6	
ALARMS:	FRAME ALARM - AFFECTED UNIT	
ACTIONS ATTEMPTED:	TDM SW	DON-2-A
	TDM SW	RMN-2-A
	TDM SW	RMN-1-A
	TDM SW	FEL-5-A
PROBABLE FAULT:	PCM	DON-2-A-6
	PCM	FEL-5-A-6

In isolating the source of a stream outage, the receive side (downstream of the failure) TCU is given responsibility. This precludes simultaneous restoral attempts by the Two TCU's

and any contention or erroneous conclusions that could result from such undiciplined action. The main reason for giving the responsibility to the receive side TCU is that, at least for unidirectional failures, this TCU still has a send capability and can exercise control over all stations in the loop. (Note that any link failure produces a service channel outage.)

Simple status changes and alarm changes that indicate failed off-line equipment are handled by the Data Acquisition software module. If the Data Acquisition software determines that the alarm change indicates a stream outage, the Automatic Fault Isolation software comes into play.

Automatic fault isolation proceeds in an orderly manner in accordance with a highly-structured algorithm. The fault isolation algorithm exploits the fact that the alarm manifestations of an equipment failure are always data stream oriented. Accordingly, the first task of the algorithm is to identify the highest order stream failure that exists. This is accomplished, in general, by the correlation of digroup and MBS associated alarm data that is transmitted from end-to-end over the exact path of the stream in stream status messages.

Once the TSC has indentified the highest order stream outage that exists, the TSC attempts to further fault isolate to the equipment level based on the existance of additional alarms or by performing a systematic remote switching of redundant equipments.

In addition to performing the functions of automatic monitoring and control the TSC is capable of functioning in other roles. Foremost of these is a manual fault isolation support function in which the TSC responds to operator requests for specific raw data and executes manual remote control commands.

SECTION 6

6.0 TELEMETRY ORGANIZATION AND PROTOCOL

This section discusses alternate approaches to satisfying the communications requirements of the transmission control subsystem.

6.1 Telemetry Subsystem Requirements

Transmission control functions that must be supported by the telemetry subsystem include data acquisition, remote control and summary report distribution

Specifically, in support of automatic fault isolation, the telemetry subsystem must remote critical alarm and status change information to the locations where this information is processed by the fault diagnosis software. In addition, the telemetry subsystem must convey specific information requests, mode change commands, control commands and command acknowledgements between the processor and the remote equipments.

In support of manual fault isolation and status reporting, the telemetry subsystem must convey equipment and alarm status from remote to manned sites for display to tech controllers. In addition, it must provide for the timely transmission of manual requests for raw alarm and monitor data, manual remote control commands and positive acknowledgement of command execution. Also, the telemetry subsystem must be capable of routing status summaries to Regional and System Level tech control facilities.

Analysis of the specifics of the subsystem requirements in the light of the Projected European DCS Connectivity - 1982 has resulted in the following list of design requirements and design goals.

6.1.1 Number of Remotes per Master Unit

The required maximum number of Remote Telemetry Units (RTU's) that must be controlled by and report to a single master Transmission Control Unit (TCU) is determined by the longest chain of repeaters between main stations or branching repeaters in the Projected Network. Examination of the network indicates that the maximum number of RTU's per TCU is six (6). In order to preclude the restriction of system growth, however, the required number of RTU's per TCU is taken to be twelve (12).

6.1.2 Number of Telemetry Points and Scan Frequency

Nearly all of the transmission equipment alarms and monitors are potentially useful for fault isolation. It is not, however, felt that the appropriate action is to routinely poll the local loop for all raw data each frame. Most supervisory control and data acquisition systems use a report-by-exception procedure. With this approach, only alarms that are in an alarming state or monitors that violate a set-point are transmitted. In order to specify the volume and rate of data that must be accommodated, it is felt that it is appropriate to assume that a report-by-exception procedure is used.

In order to size the maximum volume of data that can reasonably be expected to required collection at any one time on a single local loop, it has been assumed that a worst-case condition results for the case of a link failure where all associated digroups terminate within the same local loop. This would produce a total of at least 44 definite alarms.

It is difficult to firmly fix a requirement for scan frequency. It is taken as a goal, however, to keep the scan rate high enough so that it will not contribute objectionably to the

delay experienced in determining the success or failure of an attempted restoral action. For both the Level 1 and Level 2 multiplexers, sync acquisition is specified to take less than 50 msec. Based on these considerations, the maximum scan period is taken to be 25 msec (10 msec desired).

6.1.3 Number of Telemetry Branches per Node

(This parameter bears on two aspects of the telemetry subsystem design problem. These are: 1) the partitioning of telemetry subsystem hardware and, 2) the interloop routing of telemetry messages.)

The worst-case node in terms of number of branches is Donnersburg with thirteen (counting one cable and two FDM links). Based on this, the maximum number of telemetry branches per node that must be supported by the telemetry subsystem is taken to be sixteen (16). To permit more flexible growth, a desired design goal is modularity to provide incremental expansion beyond 16 branches. In terms of addressability (for message routing) the next logical step beyond 16 branches is 32 (5-bit branch address).

6.1.4 Interloop Routing Delay

Interloop messages are of two types: 1) regional and express traffic and 2) stream status messages. The delay encountered at each node along the transmission path is more critical for stream status messages since these are critical elements of the fault isolation algorithm. For many isolation and restoral sequences, interloop routing delay multiplied by a factor times the number of nodes traversed appears as a direct additive contributor to the isolation and restoral time.

The delay imparted to stream status messages affects two aspects of the algorithm. The declaration of a higher order stream outage based on correlation of remote digroup alarms cannot be made until the most remote digroup status is reported. On the other hand, declaration of the success of a restoral attempt requires only a report of restoral from one of the affected digroups.

Message queues can occur at network nodes giving interloop routing delay a degree of randomness. Thus, although interloop routing delay is a very important parameter, it is difficult to specify a reasonable quantitative requirement. It is important, however, to select a telemetry organization and protocol that minimizes interloop routing delay.

6.1.5 Remote Control Command/Reaction Delay

The delay between the initiation of a control command and receipt of positive acknowledgement of command execution is an important parameter. This should be minimized to provide quick response to the troubleshooting actions of the automatic fault isolation algorithm which will reduce the overall diagnosis and restoral time.

6.1.6 Degraded Operation

In the event of a service channel failure, the telemetry subsystem must detect the occurrence of the failure and automatically reconfigure the telemetry link to provide a degraded mode of operation. As a minimum, the degraded mode should provide two-way telemetry from both ends of the local link up to the point of the break.

In the event that the failure is at one terminus of the link, two-way telemetry must still exist between the other end of the link and all repeaters in the link.

Virtually 100% of all telemetry channel failures should be detectable.

6.1.7 Flexibility

The DEB network is planned to be implemented through staged upgrades. The network configuration can thus be expected to be undergoing periodic change. An important consideration in selecting a telemetry organization is, therefore, the ability to accommodate network (and local link) changes. It is expected that these changes will impact the telemetry subsystem in two ways: 1) modification of message routing tables and 2) addition or deletion of stations from local loops, e.g., addition of a repeater. It is desired that such changes have minimal impact on the telemetry subsystem.

Another aspect of subsystem flexibility is with respect to the usage of the 56 Kbps service channel resource. It is desired that the resource be used efficiently maximizing throughput for traffic supporting transmission control but at the same time permitting the simple addition of services to support other missions. As a quantitative design goal, it is desired that the TSC mission leave undisturbed a minimum of 32 Kbps of the service channel capacity.

6.2 Telemetry Concept Development

This section discusses alternative approaches to satisfying the telemetry subsystem requirements and presents the rationale for the recommended connectivity, access discipline and protocol.

6.2.1 Connectivity and Channel Access Control Alternatives

In selecting a telemetry channel organziation, there are two important and inter-related considerations: connectivity and channel access discipline.

6.2.1.1 Connectivity Alternatives

First, consider the way in which the equipment can be configured. At simple repeater sites, there are two options. These will be refered to as point-to-point and thru-connect and are illustrated in Figures 6-1 and 6-2 respectively.

With the point-to-point connection, all "direct" communications is between adjacent sites with processor handling being required for messages that go beyond one hop. With the thru-connect arrangement, hardware can be configured to simply pass-on the bulk of the traffic with the processor only "looking at" message blocks that are specifically directed to it.

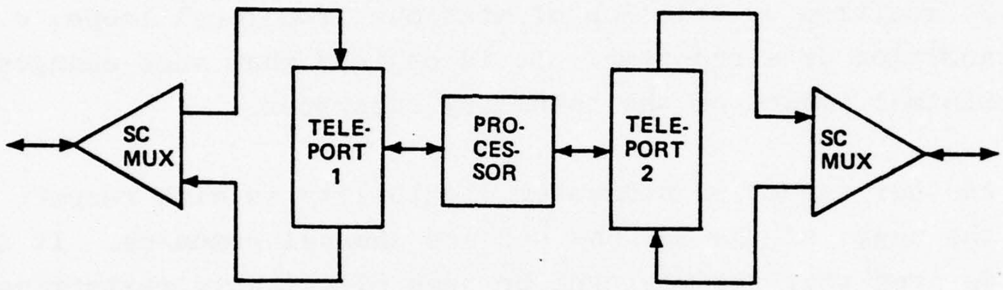


FIGURE 6-1
POINT-TO-POINT CONNECTION

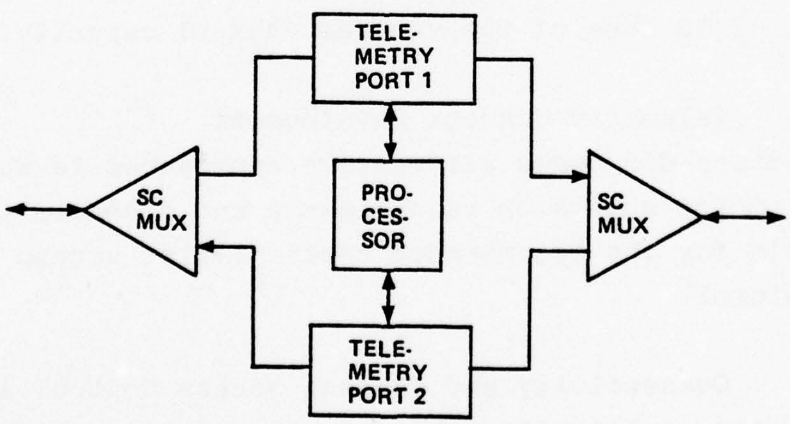


FIGURE 6-2
THRU-CONNECT CONNECTION

Local link connectivity with these two approaches is depicted schematically in Figures 6-3 and 6-4.

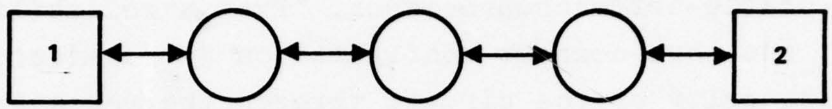


FIGURE 6-3
POINT-TO-POINT LINK CONNECTIVITY

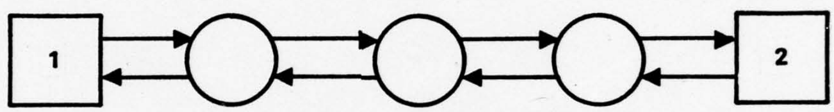


FIGURE 6-4
THRU-CONNECT LINK CONNECTIVITY

In both cases, there is a full-duplex service channel port going each way. In the point-to-point case, there are N+1 dedicated circuits with only two users where N is the number of simple repeaters in the chain. Message buffering and forwarding is controlled by the processor and since there are only two users on a full-duplex circuit, no channel access discipline is needed.

In the thru-connect link configuration, a time-division multiple access channel exists. As with any time-shared channel, a discipline must be devised by which individual users gain access to the channel.

The point-to-point connectivity is unattractive because it requires store and forward of all link messages. This means that processing resources at the remote repeater sites must be devoted to this unnecessary task. At the 56 kbps service channel rate (142 μsec per byte), this would place a not insignificant load on a state-of-the-art microprocessor. From a reliability standpoint, the thru-connect configuration is substantially better. Here, the integrity of the circuit through the repeater depends only on a shift register delay and an "AND-OR" select circuit as opposed to a complex microprocessor and its ancillary circuitry. (The simplicity of the circuitry for the thru-connect approach is indicated in Figure 6-5.)

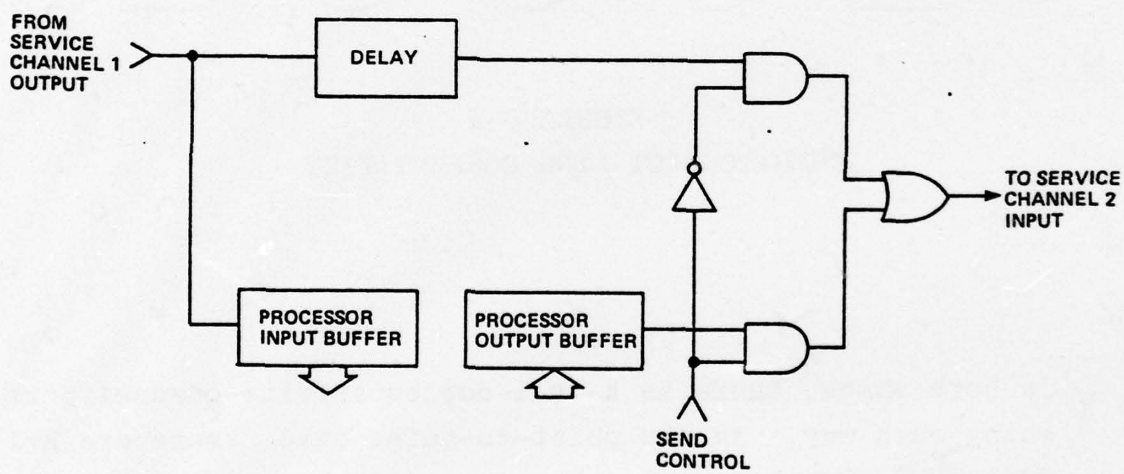


FIGURE 6-5
Repeater Thru-connect Logic

6.2.1.2 Access Discipline

A common channel, such as the thru-connect configuration, can be shared by multiple users on either a random access or controlled access basis. These two basic access disciplines are discussed below.

6.2.1.2.1 Random Access

With a random access discipline, there is no common control and any of the resource sharing terminals can initiate a transmission at will. With this discipline, several stations may transmit at once and an error control/recovery scheme is needed to combat such collisions. This technique is used in the ALOHA system and has been found to work well when the channel use factor is 18% or less of the total channel capacity. The curve of throughput versus usage has a rather sharp "knee." This is due to the fact that an increase in collisions produces an increase in retransmissions which in turn result in a further increase in collisions.

With the "chained" link connectivity of interest here the situation is somewhat different. All "upstream" transmissions pass through "downstream" stations and unilateral control can be exercised by each station to prevent collisions. Preventing collisions requires: 1) restricting message lengths, 2) an input message buffer equal in length to the maximum message length, and 3) a channel activity detector. This approach is depicted in Figure 6-6.

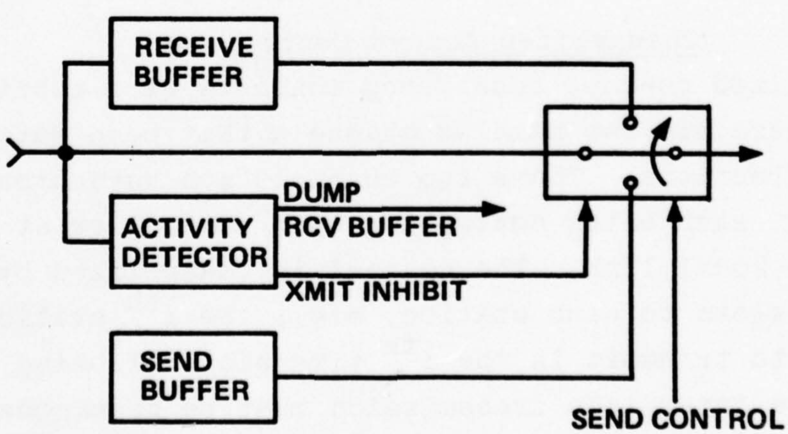


FIGURE 6-6
RANDOM ACCESS COLLISION PREVENTION SCHEME

The idea here is to prevent transmission if an upstream station is already transmitting and to buffer any upstream transmissions that start after transmission has begun. Thus, at the price of restricted message length, a buffer equal in length to the maximum message length and control circuitry of moderate complexity, collisions can be effectively prevented in the present situation where each station can exercise some control of the shared channel.

6.2.1.2.2 Controlled Access

In general, in order to efficiently utilize a common channel on a time-shared basis, some form of central control of channel access is required. Controlled access disciplines can be divided into two classes: channelized and adaptive. With a channelized discipline, a fixed allocation of resources is made to each of the several users. With an adaptive discipline, resources are allocated on an as-needed or demand basis. The adaptive approach implies, in general, some (usually small) overhead communications related to the conveying of information regarding demand and allocation.

Two approaches have been considered for configuring the TSC telemetry channel using controlled access disciplines; one channelized and one adaptive.

6.2.1.2.2.1 Channelized Access Control

The channelized concept considered consists of a configuration in which there are two simplex channels that send data in opposite directions. These two channels are asynchronous with one another, each being controlled by a controller at opposite ends of the local link. The channel is channelized by pre-assigning fixed time-slots to each station, e.g., the i^{th} station is preprogrammed to transmit in the i^{th} time slot following the frame sync. Since inter-loop transmission must be accommodated, a portion of each frame is reserved for inter-loop traffic. An example of such a channelized frame format is shown in Figure 6-7.

FRAME SYNC	SLOT 1	SLOT 2	• • •	SLOT N	INTER-LOOP MESSAGE SLOT

FIGURE 6-7
CHANNELIZED FRAME FORMAT

The 2 simplex channels will be refered to as Channel A (from Node 1 to Node 2) and Channel B (from Node 2 to Node 1). Frame sync for Channel A is sent by Node 1 and frame sync for Channel B is sent by Node 2. Two of the time slots are assigned to the nodal stations and each of the repeaters have a time-slot assigned to them. Node 1 sends remote control commands and data requests to the repeaters via Channel A; repeaters insert their responses into the next frame coming over Channel B. Interloop messages are sent between nodes in the Interloop Message Slot.

If a buffer equal in length to the repeater time-slot is provided at each repeater, repeater responses can occupy the same time slot that is used to send remote control commands to the repeater. This is illustrated in Figures 6-8 and 6-9.

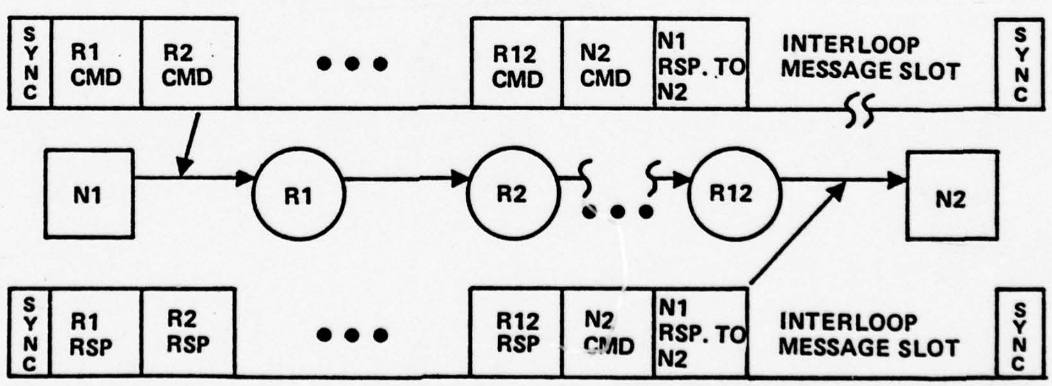


FIGURE 6-8
CHANNELIZED POLL/RESPONSE SCHEME

With this arrangement, there is a one time-slot delay through the RTU. The Sync Detector and timing logic determine when the Receive Buffer is fully loaded with the poll message addressed to this station. At this instant, the switch is thrown connecting the send Buffer to the output. At this instant also, the Receive Buffer is read. The Receive Buffer continues to be clocked while the Send Buffer is being read-out. Thus, as soon as the last bit of the transmitted message is clocked out of the Send Buffer, the first bit of the next succeeding time-slot is available at the Receive Buffer Output. At this point, the timing logic returns the transmit switch to its normal position.

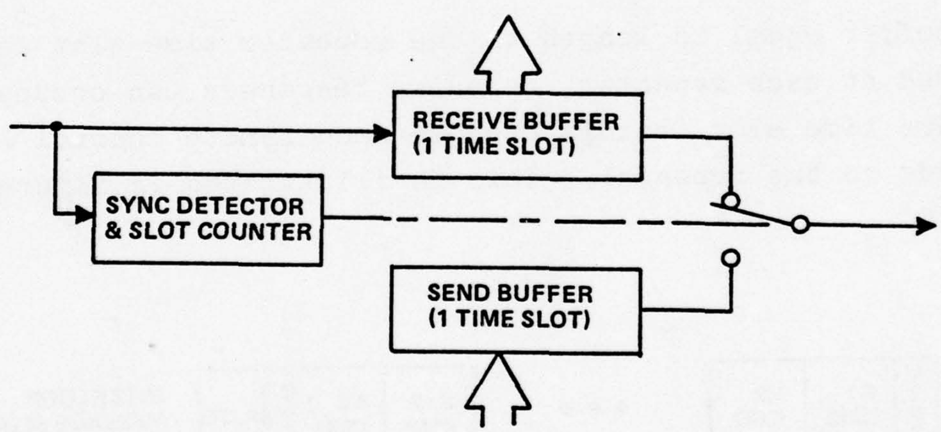


FIGURE 6-9

TELEMETRY INTERFACE FOR CHANNELIZED ACCESS

With this scheme, time slots are fixed in length. In order to provide a relatively quick response time, the frame, and hence individual time-slots, must be kept fairly short. This can be done at the expense of a small amount of overhead. (In general, response messages will occupy multiple frames).

The actual time slot size used to determine maximum scan frequency was determined based on: 1) a desire to convey all the information required by a remote control command in a single frame; 2) a desire to include sufficient information in a slot to keep the overhead from being unreasonably high; and 3) a desire to keep the slot as short as possible in order to keep the frame time short. It should be kept in mind that the overhead penalty is not a very important consideration if reporting is by exception. This is due to the fact that data heavy time-slots will be very infrequent. If this is the case, there is no point in making the time slots any longer than is needed to accommodate the remote control command structures. If longer time slots were used, many time-slot bytes would normally be unused.

6.2.1.2.2.2 Adaptive Access Control

A particular form of adaptive access control that is attractive from a number of standpoints including performance and ease of implementation has been investigated in some detail. This approach makes use of a 'loop' type of connectivity. The DEB network service channel resource lends itself quite well to loop connectivity. A local link configured as a loop is shown in Figure 6-10.

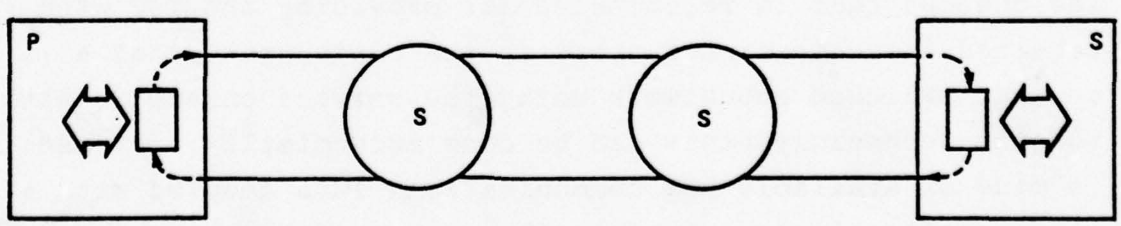


FIGURE 6-10
LOOP CONNECTIVITY

Here, one of the two nodes is designated as the loop Primary station. The repeaters and the other node in the loop are designated as Secondary stations.

Data loops have been studied by a number of investigators [2-4]. The key feature of the loop configuration is its inherent direct feedback. Because of this characteristic, the loop lends itself to adaptive access control disciplines.

In the loop configuration, channel access is controlled by the circulation of a "GO-AHEAD" character. Transmission proceeds as follows. The primary sends its data followed by a "GO-AHEAD." The first down-loop secondary station with the authority and need to transmit does so upon detection of the GO-AHEAD. It does so by suspending the repeater function, destroying the existing GO-AHEAD, sending its data and appending a new GO-AHEAD. As the GO-AHEAD propagates around the loop, each station in turn has an opportunity to transmit. When the GO-AHEAD has propagated back to the primary, the cycle starts over again. The key point of this scheme is the fact that the feedback enables the start of a new frame to be triggered by the end of the preceeding frame. The frame length thus automatically adapts to the communications load.

6.2.2 Recommended Concept

The concept that is recommended for providing the TSC with required communication support is the configuration of a message switched subnetwork using the service channel. With today's technology, this can be done economically. If use is made of available LSI communications IC's coupled with a microcomputer implementation, the cost difference between a primitive telemetry system and a flexible, message-switched network is not very great.

The service channel subnetwork concept is illustrated in Figure 6-11. Local loops implementing a simple "GO-AHEAD" access discipline serve to interconnect network nodes. A link-level protocol governs the transmission of information on these local loops.

Message packets sent over local loops are routed toward their destination at network nodes by the TCU processor. A network-level protocol governs the end-to-end message exchange.

It should be noted that while much of the TSC mission traffic is confined to the local loop, interloop data transmission is essential to the fault isolation mission of the TSC. This is because of the fact that data streams generally span several local loops and the remoting of stream related alarms requires interloop data transmission.

The loop type of connectivity and access discipline has a number of advantages for the TSC application. In the local loop concept, the two TCUs at each end of the loop have the responsibility for monitoring all of the other stations on the loop. If a report by exception scheme is adopted, the primary circulates polling messages but responses are only generated when a station detects a change in one or more of its critical alarms. Thus, in the normal quiescent state, the primary generates all-call polls one right after another and scanning of the stations on the loop for alarm changes is very rapid. The TCU's at either end of the loop receive all message blocks whereas the repeaters have selective address detection circuitry so that they "look at" only all-call or uniquely addressed polls. Repeaters and the TCU functioning in a secondary role repeat all message blocks while the primary acts as a sink for all message blocks coming back over the link.

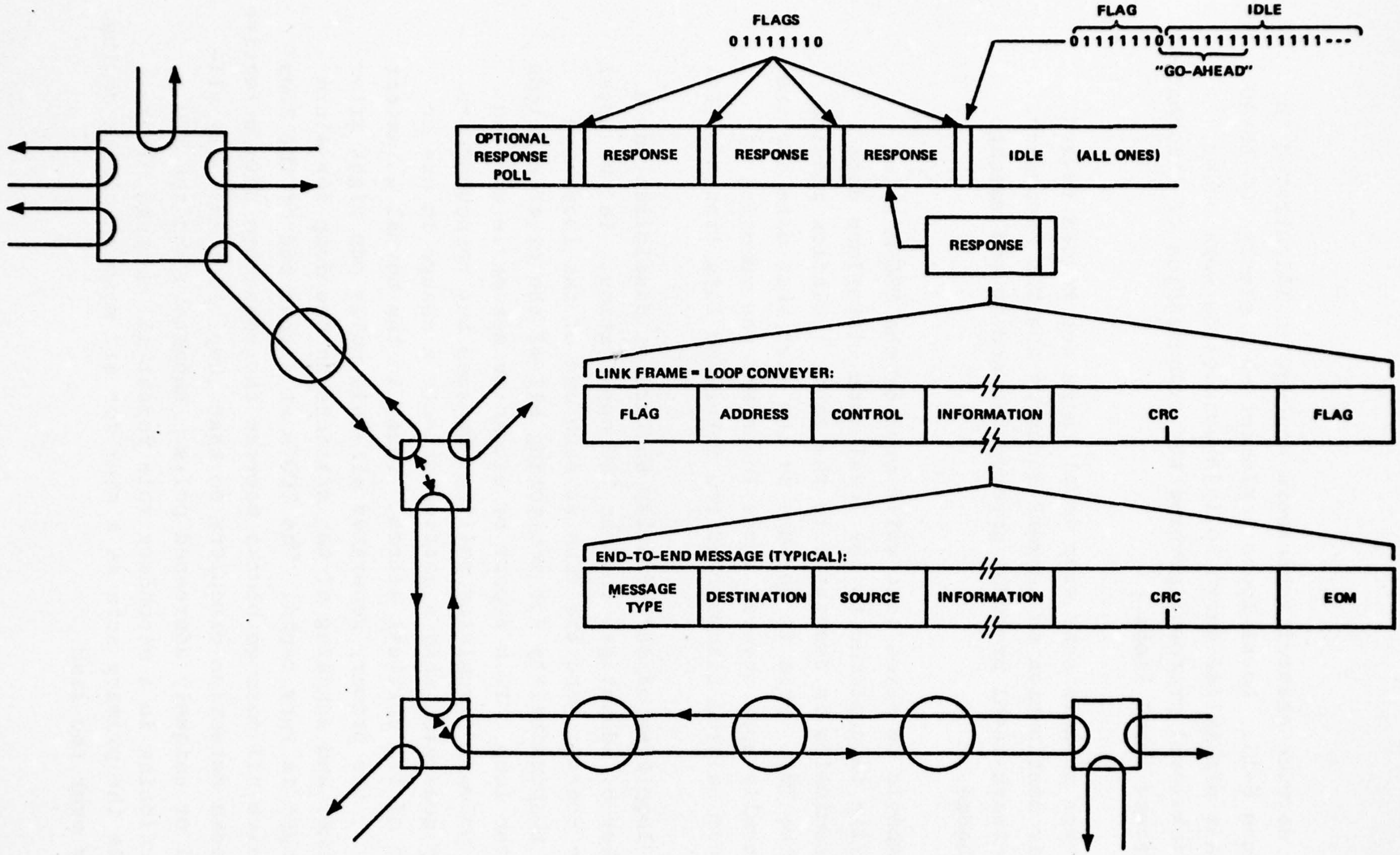


FIGURE 6-11
MESSAGE SWITCHED SUBNETWORK

The TSC data communications subsystem is a message switched subnetwork. The concept provides buffering for interloop messages at each node. Via the TSC telemetry subsystem, messages of any length (blocked into frames of suitable size) can be transmitted between any two points in the network. When an interloop message frame appears at a nodal buffer, it is sent out on the appropriate branch appended to the normal routine poll for alarm changes. In this manner, the resource is demand assigned to the needs of interloop communications.

The percentage of the resource that is shifted to interloop communications when the demand arises is controllable by the system software. If there was a need to push through a very high priority message type, it would be possible to suspend routine polling altogether and devote the entire 56 Kbps (minus some small overhead) to the transmission of these messages. In the absence of such priority traffic, it is proposed to adopt a simple scheme that will allow a single interloop message frame to be inserted between each poll. Variable length message frames are permitted subject to some appropriate maximum determined by the channel error rate and TCU buffer size. Based on previous work it appears that a maximum frame length on the order of 1000 bits is appropriate.

The flow of message frames over the loop for various situations is illustrated in Figure 6-12.

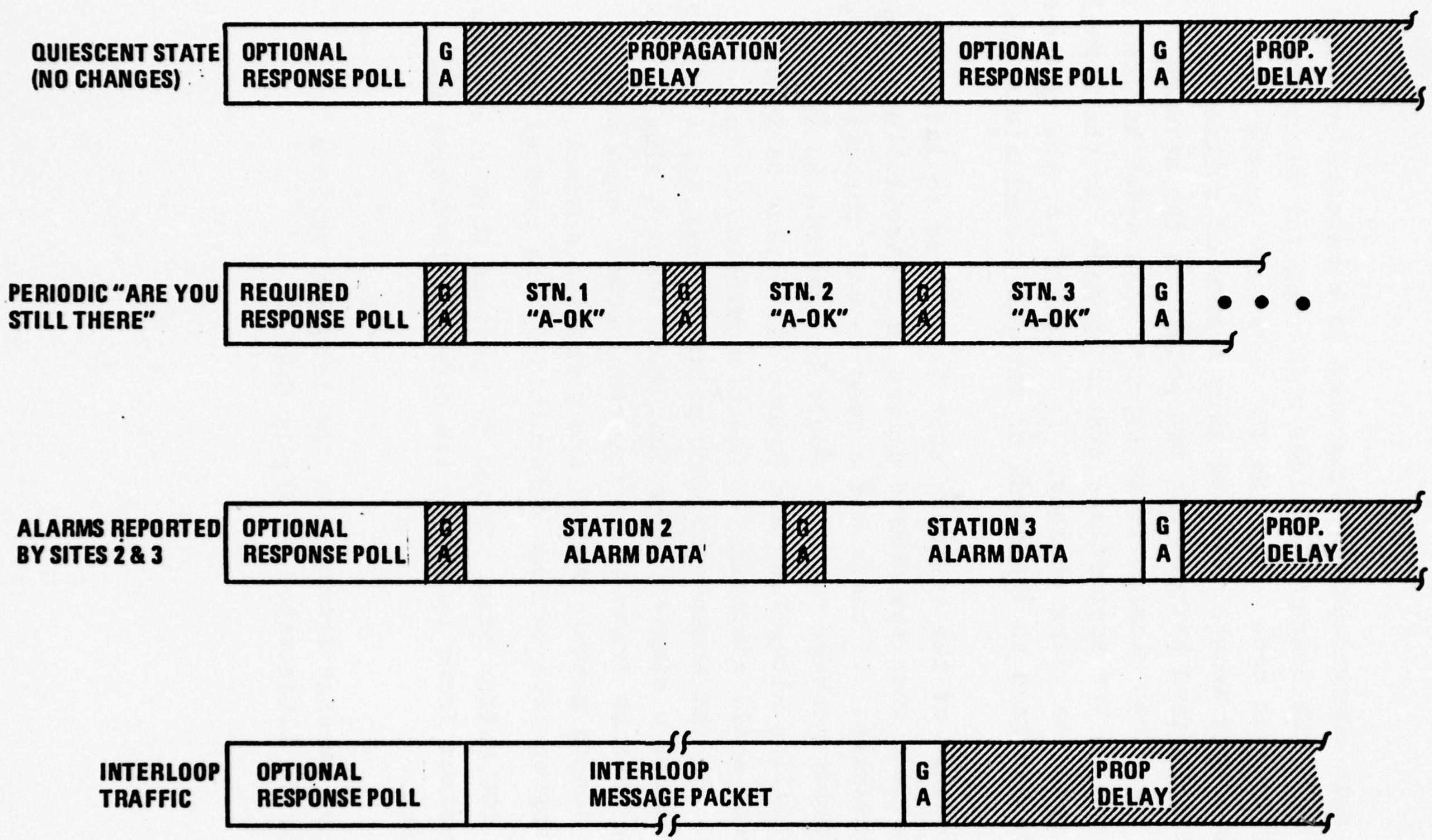


FIGURE 6-12

LOCAL LOOP MESSAGE FLOW

In addition to the inherent adaptive allocation of the resource, this loop transmission scheme possesses another advantage that is equally important in the TSC application. This is the fact that station addressing and access is not based on positional information within some pre-defined frame. There is no required count-down to determine the station's access slot; a station's "ticket" to get on the loop is simple GO-AHEAD recognition. This means it is extremely simple to make changes in the configuration of a loop; for example, to add a repeater.

6.2.2.1 Communications Protocols

A communication protocol is really a specification for an information transfer procedure. Historically, in telemetry systems involving a simple point-to-point channel, custom formats and procedures were devised to meet the end-to-end data transfer requirement. In a complex network, a data transfer really involves a nested hierarchy of data transfers: user-to-user; source-to-destination; adjacent node-to-adjacent node.

Concerted efforts toward standardizing data communications protocol are currently underway by ISO, CCITT, ANSI and EIA. IBM has developed their own "standard" data communication procedures called SNA (System Network Architecture) incorporating its widely known link protocol, SDLC (Synchronous Data Link Control). All of these efforts have recognized the desirability of separation of functions with the goal of producing standards that are non-restrictive, i.e., the development of a protocol hierarchy in which one level does not impose requirements on another.

There are at least two good reasons why the TSC telemetry subsystem protocol should be chosen in accordance with these newly developing standards. First, the relative

independence aspect of these proposed protocols is important in the TSC application. With this attribute, the telemetry subnetwork will have the generaltiy it needs to support not only the TSC mission but other systems communications requirements as well. Distinct mission traffic with distinct user-to-user protocols can efficiently coexist on the network sharing common link and network protocols. Secondly, semi-conductor houses are already beginning to respond to these coming standards by offering LSI multi-protocol USRTs (Universal Synchronous Receiver/Transmitters) that support many of the functions of the coming link-level protocols (ADCCP, HDLC and SDLC).

The recommended communication protocol for the DEB telemetry subnetwork thus involves a hierarchy of protocols: user-level, network-level and link-level.

This hierarchy is illustrated in Figure 6-13. In this hierarchy, the function of link-level protocol is to reliably deliver network message blocks between network nodes. The responsibility of network-level protocol is the reliable delivery of user message blocks between the source and destination terminal equipments. The function of user protocol is to support the information transfer needs of the man or the applications programs that are using the system.

In the same way that the several levels of protocol are nested in the physical network, they are also generally nested positionally within the transmission frame. This is illustrated in Figure 6-14. It should be noted that, in certain cases, one or more of the protocol layers may not be required. A case of this in the TSC application is where

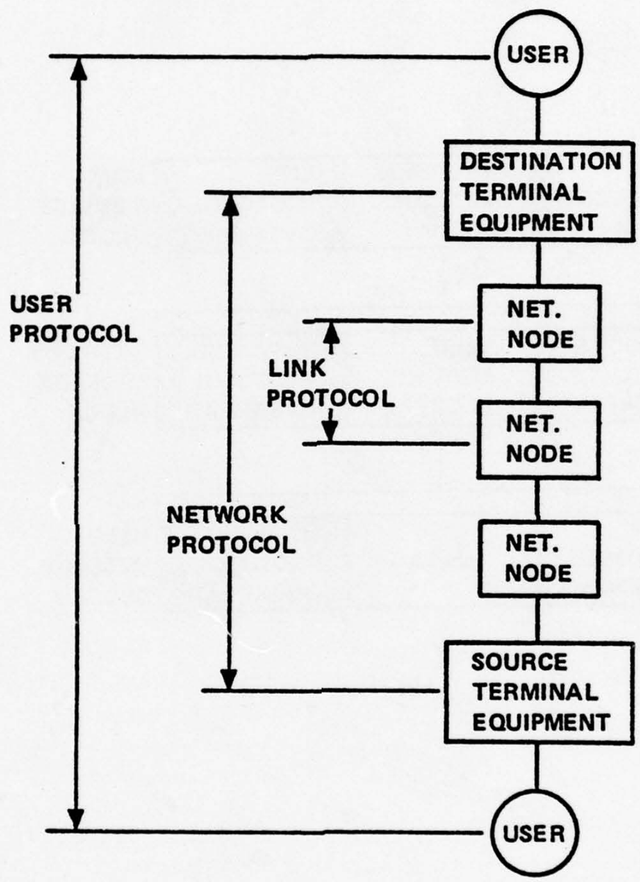


FIGURE 6-13
PROTOCOL HIERARCHY

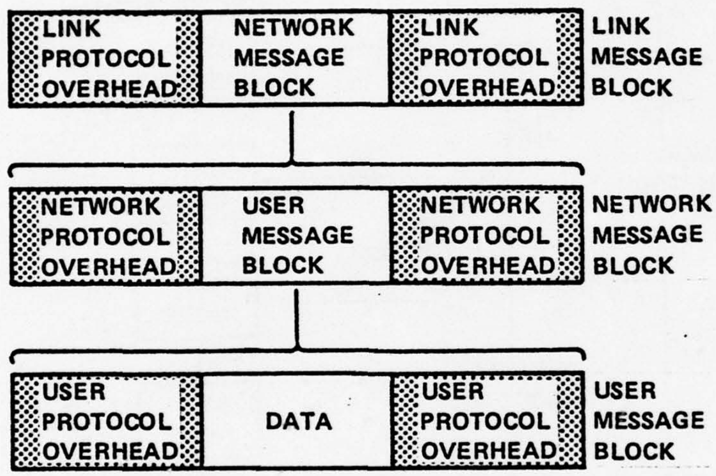


FIGURE 6-14

HIERARCHICAL STRUCTURE OF MESSAGE BLOCKS

the information transfer is between a source and destination that lie on the same local loop. An example is a status change message sent from an RTU to the associated TCU. Here, none of the functions of network-level protocol are needed. In the case of an ACK/NAK message, none of the functions of user protocol are required. Link protocol functions are required for any information transfer.

One of the main goals of structured protocol is the separation of functions. Each layer addresses only the functional needs of that layer. By maintaining this functional independence, transparency is achieved. Each layer is capable of stand-alone interpretation, e.g., the meaning of a user message block does not depend on a control byte that is part of the network protocol. This permits complete freedom in the sharing of the communications resource by different missions. A user protocol that is presently defined to support the TSC mission places no constraints on the definition of a user protocol that may be devised in the future to support some other mission. The coexistance of multiple mission traffic in the network will cause no complications.

The conclusion of this study is that the structured approach should be adopted in designing a communications protocol for the DEB telemetry subnetwork. Many of the details of the complete protocol are not addressed in this report.

The user-level protocol requirements of the TSC mission have been defined. Formats are given in Section 8 .

The general requirements for network-level and link-level protocol are defined. The link-level message block format is presented. Many of the details of link and network protocol procedures do not require definition at this

time. These need further study in the light of new standards that are being developed by ISO, CCITT and ANSI.

Within each hierarchial level, there are two considerations to be addressed in providing a complete specification: formats and procedures.

6.2.2.1.1 Link-Level Protocol

Link-level protocol deals with the transfer of data between adjacent nodes of the network. Of the three protocol levels, this level has received the most attention to date and is the closest to being standardized. It is this level also that is currently receiving the attention of the designers of general-purpose LSI.

There are two link-level protocols that are currently under consideration by standards organizations: HDLC and ADCCP. A third protocol, SDLC, is meanwhile rapidly becomming a de facto industry standard in the commerical computer-to-computer communication field. There is a great deal of similiarity between HDLC, ADCCP and SDLC. All have identical message block formats. This format is shown in Figure 6-15. All are bit oriented protocols and use a unique flag word to delimit the message block.

This high degree of commonality has made it possible for a number of semiconductor houses to develop LSI multi-protocol universal synchronous receiver/transmitter (USRT) devices that can be programmed to support any of these link protocols. A block diagram of one such device (SMC COM5025) is shown in Figure 6-16. These devices permit the off-loading of a great many (but not all) of the link-level protocol functions

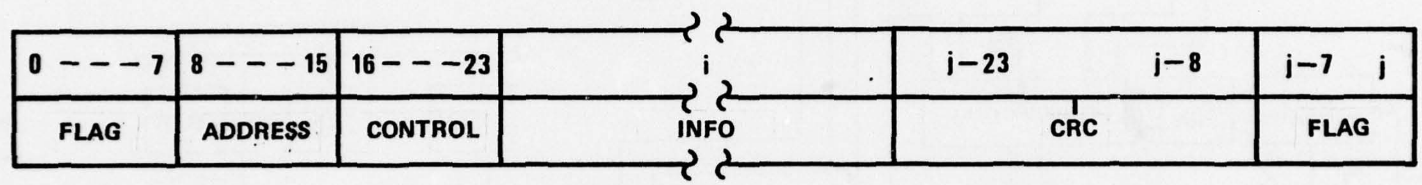


FIGURE 6-15

STANDARD LINK-LEVEL PROTOCOL FORMAT

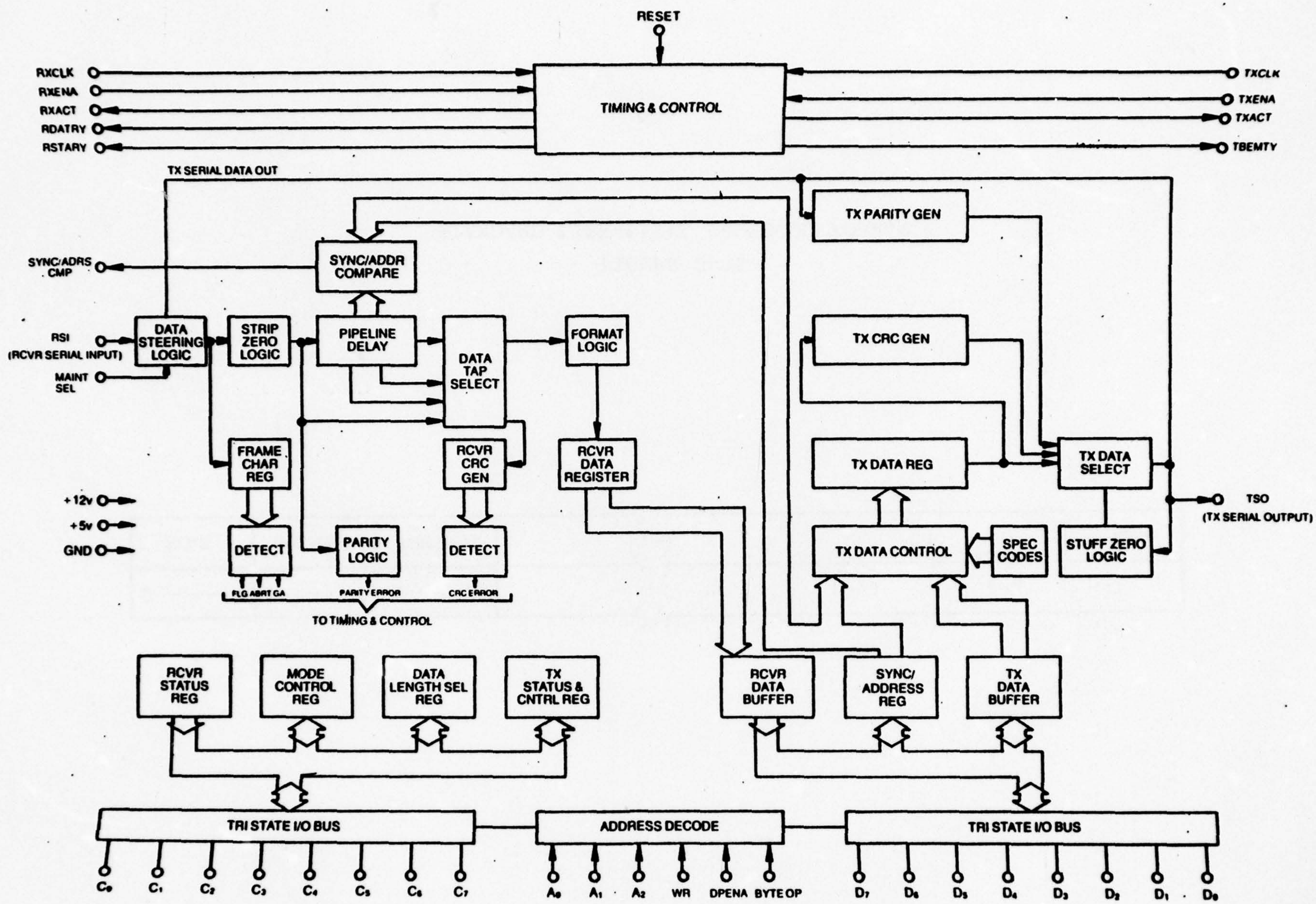


FIGURE 6-16

MULTIPROTOCOL USRT BLOCK DIAGRAM

from software to hardware. This takes a substantial load off the telemetry or host processor especially when data transmission rates are moderately high. Functions typically performed by these devices include:

- Flag Character Generation/Detection
- Bit Stuffing/Stripping
- Address Match Detection
- CRC Generation
- CRC Error Checking
- Idle Pattern Generation
- Detection of Loop "Go-Ahead" Character
- Processor Handshaking

Link related protocol functions that still must be done in software include keeping track of message block sequence numbers, ACK/NAK generation, error recovery, mode control and initialization.

All fields shown in the format of Figure 6- 15 except the information field are required in all transmissions. All of the fields with the exception of INFO (variable length) and CRC (2-bytes) are single 8-bit bytes. All transmission blocks (frames) are thus 48 + i bits in length were 'i' is the length of the information field. While the information field is variable in length, some practical limit is in order based on operational and error control considerations.

Although frames can occur asynchronously and are of variable length, the bit content of the frame itself is isochronous.

Since the start of any frame cannot be predicted and frames can be of variable length, some means of flagging the start and end of each frame is required. This is done by using a unique sequence called a FLAG at the beginning and ending of each frame. This sequence (01111110) is prevented from occurring in the remainder of the body of the frame by a zero insert/delete algorithm that works as follows.

When transmitting:

A station monitors the sequence of bits transmitted between flags. If a sequence of five contiguous 'ones' is detected, the transmitting station automatically inserts a binary zero into the information stream. As a result of this zero-insertion, no more than five contiguous 'ones' will be transmitted within the frame. In this manner, FLAG characters are prevented from occurring in the address, control, information and block check fields.

When receiving:

A station inspects the bit following any occurrence of five contiguous 'ones.' If this bit is a 'zero,' the receiving station deletes it from the information stream. If the bit in question is a 'one,' the sequence is either a FLAG character or an error. When a sixth 'one' bit is received, the station examines the next received bit. If this bit is a binary zero, the sequence is accepted as a terminating FLAG; if the bit is a binary one, the frame is rejected.

The first eight bits following a beginning FLAG contain the secondary station address and are included in all frames from both primary and secondary stations. In transmissions to secondary stations, the field designates which secondary station (or stations in the case of a group or broadcast address) is to receive the frame. In transmissions from a secondary station, the field designates the secondary station from which the frame originated. The address is handled as an eight-bit entity and may be used to refer to a single station or a group of stations.

The second byte following a beginning FLAG is the control field. The purpose of the control field is to convey control information relating to modes, procedures and information transfers. The details of the control field are not discussed here because they are quite involved, they vary among the 3 candidate protocols, and it is premature to select one of the three at this time.

For the purpose of this study, it is assumed that the chosen link-level protocol procedures will provide for centralized link control, i.e., secondary stations on the loop transmit only when polled by the primary. It is also assumed that the chosen protocol will make provisions for an "optional response poll" and a "required response poll" to be used in the task of local loop data acquisition.

An information field is not necessarily included in all frames. When present, the information field immediately follows the control field and continues up to, but does not include, the block check field. The length of the information field is restricted only by buffering constraints of the stations involved in the information transfer and by the usual considerations of transmission block length due to communications channel error characteristics.

The information field may contain any bit sequence configuration (i.e., full transparency is the normal condition) to convey header information, control information, status, text (user data), etc. The content of the information field should be defined by actual or implied information included in the frame.

All link frames include a block check (BC) field for the purpose of detecting errors that may occur during transmission. The checking is based on the transmission of redundant information in the form of a remainder polynomial numerator R derived from a division of the transmitted data by a generator polynomial; that is:

$$\frac{P}{G} = Q + \frac{R}{G}$$

where

P is the transmitted data polynomial
G is the fixed generator polynomial
Q is the whole polynomial quotient
R is the remainder polynomial numerator

The checking accumulation is initiated by the first bit following the beginning FLAG and includes all bits up to, but not including, the ending FLAG <u>except</u> those zero bits inserted by the transmitter and deleted by the receiver as a result of the occurrence of five contiguous one bits in the transmitted bit stream (to prevent unwanted FLAGs).

6.2.2.1.2 Network-level Protocol
This report treats only the aspects of network-level protocol that are needed for the TSC mission as outlined in Section 4.0, Operations Concept. The development of a comprehensive specification of network-level protocol is left for future study. It is recommended that such study begin with a review of the proposed standards being developed by ISO, CCITT and ANSI as they relate to network-level protocol.

A minimum set of functions that must be provided by network-level protocol include: addressing, routing and flow control, packet sequencing, message sequencing, message delimiting, error control/recovery and mode control.

Two kinds of interloop message are required for the TSC mission. The main characteristic that sets these two message types apart is the routing algorithm applied to each. Critical to the fault isolation algorithm are "Stream Status Messages." The purpose of these messages is to identify stream outage and restoral events to TCUs along the path of the stream and to convey outage notifications along stream paths all the way to digroup end points. The information that these messages carry is of interest not only to the ultimate destination but to each TCU along the route of the stream. Accordingly, these messages must be routed along the exact paths followed by network digroups. This means that at each node stream status messages are routed in accordance with a stored local digroup connectivity table. (A method for efficiently storing digroup connectivity is presented in Appendix B.) One possible format for stream status messages is given in Section 8. This format illustrates the information that is requisite in a stream status message. It was not developed according to any generalized network-level protocol structure.

For ordinary interloop message transmission, a less restrictive routing algorithm is appropriate. Furthermore, unlike stream status messages, a general interloop message can span several transmission frames (be comprised of a sequence of packets). Since each individual packet is handled independently at each node, it is possible that packets transmitted in sequence

can arrive in a different order than the one in which they were sent. Because of this, a packet numbering system is required. A numbering system for messages sent between a particular source/destination pair is also desirable. The details of these network protocol procedures are not addressed here. These should be specified after a thorough review of the cited forthcoming standards. One observation, however, is made with regard to the general routing algorithm.

It is suggested that general interloop messages be routed in accordance with routing tables stored at each node that specify both a primary and an alternate route. The link connectivity generally permits this and the inclusion of the alternate route will enhance transmission reliability. The memory requirement for storing such a routing table is modest. For example, assuming 16 branches at a node and a total addressability of 256 stations, the required memory capacity would be 2048 bits. Table 6-1 illustrates the memory organization for a routing table for both primary and alternate routes

STATION ADDRESS	PRIMARY BRANCH	ALTERNATE BRANCH
00000000	0010	1100
00000001	0010	0110
.	.	.
.	.	.
.	.	.

TABLE 6-1
Routing Table Organization

6.2.2.1.3 User-level Protocol

The standardization of user-level protocol has, to date, not been addressed by any of the organizations responsible for communications standards. This is due, in part, to a belief by these committees that user-level protocol is not really a function of the communication system. It is more properly considered to be within the domain of the data processing system. Secondly, because of the diversity of information transmission needs, standardization may be nearly impossible. Finally, it seems questionable that there is really any value in standardizing user-level protocol. At least in an isolated, special-purpose system, there seems to be no reason why a custom protocol should not be adopted. For the TSC, this is the approach that has been taken.

In devising a user-level protocol, consideration must be given to all of the following message types. A letter code is given after each message type that is anything other than a general type of network message (L ≡ confined to local loop; S ≡ routed over exact path of an associated data stream).

1. Routine Poll (L)
   a) Optional Response Poll (L)
   b) Mandatory Response Poll (L)
2. Raw Data Request
3. Initialization Request (L)
4. Data
   a) Change Report (L)
   b) A-OK Message (L)
   c) Raw Data
5. Mode Change (L)
6. Initialization Response (L)
7. Device Select
8. Control Command
9. Control Acknowledge

10. Free Text
11. Software Download
12. ACK/NAK
13. Stream Status (S)
    a) Stream Alarm (S)
    b) Outage Notification (S)
    c) Restoral Notification (S)
14. TSC Control Handoff (S)
15. Walburn Bypass
16. Bypass Confirmation Request
17. Bypass Confirmation

Preliminary formats for some of these message types are given in Section 8.5. Further work is needed to provide detailed format specifications for all of the message types listed.

6.2.2.2 Service Channel Failures

The problem of service channel failures was considered for two different RTU telemetry interface configurations: one in which the RTU had a single telemetry interface port and one in which telemetry ports were provided on both of its thru-paths. It was concluded that, in order to adequately respond to all of the various service channel failure conditions, the dual-port configuration was essential. This configuration is illustrated in Figure 6-17.

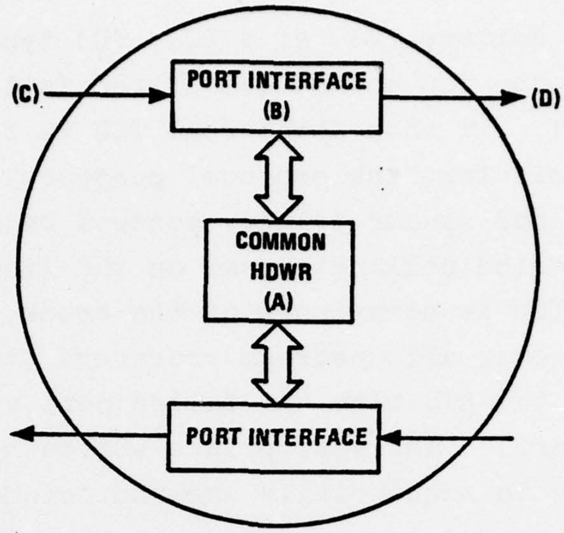


FIGURE 6-17
RTU CONFIGURATION

In considering service channel failures, there are three cases that require consideration. These include: (A) RTU common hardware; (B) RTU port hardware and (C),(D) radio or service channel mux failures (letter designators refer to Figure 6-17).

In the case of an RTU common hardware failure, reconfiguration under external control cannot be relied on since communications may be entirely out. The extent of the RTU telemetry related common hardware is simply the processor itself and the I/O bus. Most failures involving this hardware will cause the watchdog timer (see Section 11) to time-out. This can be used to derive a control signal to effect the required reconfiguration.

A port interface failure (B), or a (C), (D) type failure is detectable by the TCU downstream of the failure based on loss of signal. If this downstream TCU is functioning in a secondary role for link protocol purposes, it must flag the failure and assume primary control by placing a special, fixed-period polling signal on the line. Since the responsible TCU is downstream of the break, it can still exercise control over all upstream repeaters (it can communicate with the RTU with the failed port via the remaining good port). The appropriate action by the downstream TCU is to sequentially command telemetry port bypass and receiver and transmitter switching at each successive repeater until it detects the return of its signal indicating restoral. The telemetry port bypass circuit is shown in Figure 6-18.

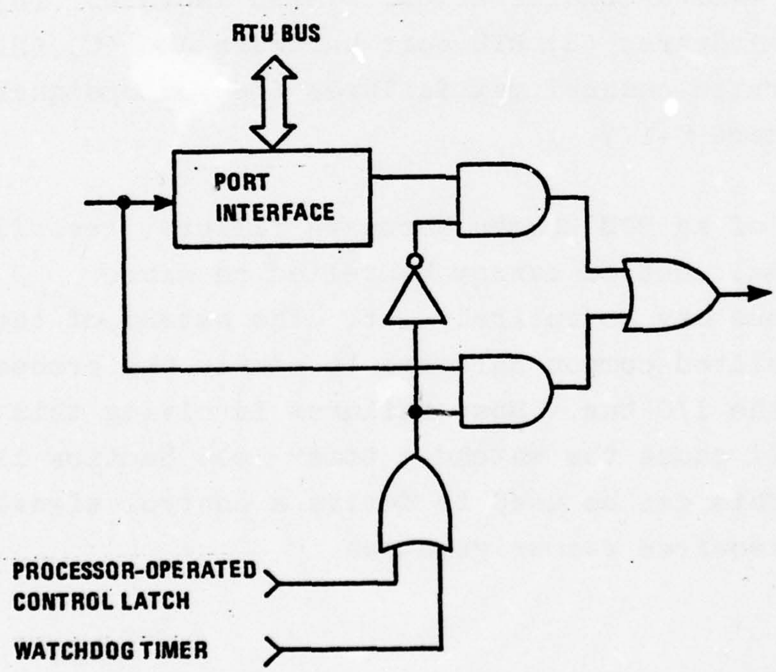


FIGURE 6-18
TELEMETRY PORT BYPASS

SECTION 7

7.0 ANALYSIS OF FAILURE MODES AND SYNDROMES

One of the missions of the TSC is to perform alarm correlation and automatic fault isolation in an attempt to restore digital stream outages. Fault isolation is, of course, only pertinent to failures that are unalarmed by the failed equipment (if the source of the failure is explicitly alarmed, no isolation is needed). Likewise, with one possible exception*, fault isolation does not apply to standby redundant equipment. If the failed standby is alarmed, no isolation is needed; if not, there is no basis for discovering the failure because there are no upstream or downstream evidences.

Failures, then, are of two types: explicitly alarmed and inferred. Explicitly alarmed failures are handled by the data acquisition and report generation software. The fault isolation algorithm is not involved.

The fault isolation algorithm deals with unalarmed, on-line unit failures. Unalarmed, on-line equipment failures by definition produce "stream" outages. Four kinds of streams are distinguished: link, MBS, digroup and VF channel. It is

---

*The cited exception is the idea of performing routine switchovers of redundant units in an effort to discover unalarmed standby unit failures before a catastrophic double failure occurs. If the standby should be in a failed condition, when it is switched on-line this fact would immediately be discovered by the appearance of downstream alarms. Detection of these alarms would immediately cause the failed unit - its failed state now discovered - to be switched back off-line. The penalty thus incurred is a very short outage compared with the much longer outage that would result if the failure went undiscovered until the on-line unit failed.

the function of the fault isolation algorithm to examine the alarm set (syndrome) and, by correlating these alarms, to infer the highest level stream outage that exists. The algorithm's task is then to attempt to isolate the failed equipment and restore service by redundant equipment or bypass switching.

In general, the syndrome does not uniquely identify the failure, i.e., a particular syndrome could be the result of any one of a number of possible causes. Diagnosing the failure consists of observing the syndrome, identifying a list of possible causes and then attempting to identify the actual cause from the list of possible causes.

7.1 Purpose and Scope of Failure/Syndrome Matrix

In order to identify and provide a tabulation of the set of possible causes associated with each syndrome of interest, a matrix of failures and resultant syndromes was generated.

Initially, the problem of generating a failure/syndrome matrix was approached by picking a specific, representative network segment as an analysis model, postulating the occurrence of the important types of equipment failures and tabulating the alarms that would have resulted at the various stations of the model. In doing this, it was quickly recognized that the key to fault isolation is the correlation of data stream related alarms, i.e., fault isolation is circuit oriented. This implies that as long as adaquate support telemetry capable of remoting digroup and MBS stream status is provided, an idealized network segment analysis model is not needed (and neither is it appropriate for algorithm development). Regardless of where a digroup or

MBS terminates, its status is visible to all interested TCUs. The appropriate failure/syndrome matrix for algorithm development is of a generalized nature since the algorithm must address all network segment configurations. The generalized matrix is a tabulation of important failure modes together with the syndrome that occurs downstream (output side of) and upstream of the failed equipment.

The failure/syndrome matrix is organized by outage type which is defined by the type of "stream" that is affected. Four stream types are distinguished: link, MBS, digroup and VF channel. Each stream is made up of a confluence of one or more streams of the next "lower order." When a stream outage occurs, outages are experienced on all associated lower order streams. Because of this, the alarm set comprising a lower order stream syndrome is a subset of each higher order stream syndrome.

The form of the generalized failure/syndrome matrix is shown in Table 7-1. The symbolism: $\{A\}_i^u$, denotes a set of alarms or syndrome. The manner in which this matrix is generated is to postulate, for each outage type, each of the potential causes and to tabulate the resultant downstream and upstream syndromes. Once this has been done for all possible causes, the potential causes are grouped according to like syndromes. The aim of this is to identify bases for discriminating between sets of possible causes and to generate troubleshooting action sequence lists for use in fault isolating.

7.2 Failure/Syndrome Matrix for DRAMA/DEB

The TSC processor software responds to changes that are detected in equipment alarms and monitors. Not all changes

TABLE 7-1

FORM OF GENERALIZED

FAILURE/SYNDROME MATRIX

FAILURE TYPE	POTENTIAL CAUSES	UP-STREAM SYNDROME	DOWN-STREAM SYNDROME
LINK	I.a) P.C. #1	$\{A\}^u_i$	$\{A\}^d_i$
	b) List of causes that produce		
	c) same syndrome as (a).		
	⋮		
	II.a)	$\{A\}^u_{ii}$	$\{A\}^d_{ii}$
	b)		
	⋮		
MBS			
DIGROUP			
VF CHANNEL			
SERVICE CHANNEL			

result in the execution of the fault isolation algorithm. Three classes of changes are distinguished: simple status changes, alarm changes that unequivocally indicate a failure in the alarming equipment, and alarm changes that could be the result of a failure in another piece of equipment. The fault isolation algorithm is concerned with only the latter and the failure/syndrome matrix deals only with this class of syndrome. Note that, because of this, those DRAMA alarms that would unequivocally flag the failure do not appear in the failure syndrome matrix. (The failure/syndrome matrix has application only in cases where uncertainty is involved.)

The DRAMA/DEB network failure/syndrome matrix is given in Table 7-2. The assumed equipment configuration for the purpose of development of the matrix is shown in Figure 7-1. In the figure, "⁓ʃ ʃ⁓" is intended to indicate possible remoteness. As previously pointed out, for all of the listed equipment failures, it should be understood that this means the on-line unit and, in every case, the failure is unalarmed by the failed equipment.

At this stage, it is necessary to postulate failures in a somewhat gross manner. In the development of the failure/syndrome matrix, it has been assumed that the failure mechanism is such as to render the affected streams in some sense "bad." The definition of "bad" is limited to the postulates that a bad stream produces frame sync alarms and an increase in BER pulses in subordinate demultiplexers if the failure is in a multiplexer or the Walburn (cases where only baseband signals are affected). If the failure is between the input to the transmitter final and the point

TABLE 7-2 DEB FAILURE/SYNDROME MATRIX

FAILURE TYPE	POSSIBLE CAUSES	DOWN-STREAM SYNDROME (OUTPUT SIDE)	SYNDROME	REMARKS
(1) Link Data Failure (Good Sync)	I a) Radio TDM MUX Common Equipment b) Radio TDM DEMUX Common Equipment	AO–Service Channel FA Telemetry Alarm PS–Level 2 TDM FA PS–Level 1 TDM FA PS–Level 2 BER Pulses PS–Level 1 BER Pulses	PS–Level 1 TDM CGA Telemetry Alarm	Service Channel Outage
(2) Link Data Failure (Bad Sync)	I a) Radio TDM MUX Common Equipment b) Radio TDM DEMUX Common Equipment	Above + AO–Radio TDM FA AO–Frame BER-Radio	As Above	Service Channel Outage
	II a) Modulator b) Transmitter (Good Final) c) RCVR (IF or Baseband) d) Demodulator	Above + AO-SQM	As Above	Service Channel Outage
	III a) Transmitter Final b) RF Fade c) RCVR Front-End	Above + AO–RSL	As Above	Service Channel Outage

ABBREVIATIONS:

AO - Adjacent only
PS - Possibly scattered
FA - Frame Alarm
SQM - Signal Quality Monitor
RSL - Received Signal Level
CGA - Carrier Group Alarm
SC - Service Channel

TABLE 7-2 DEB FAILURE/SYNDROME MATRIX (Cont)

FAILURE TYPE	POSSIBLE CAUSES	DOWN-STREAM SYNDROME (OUTPUT SIDE)	SYNDROME	REMARKS
(3) MBS Data Failure (Good Sync)	I			
	a) Level 2 TDM MUX Common Equipment	PS–Level 1 TDM FA	PS–Level 1 TDM CGA	
	b) Level 2 TDM DEMUX Common Equipment			
(4) MBS Data Failure (Bad Sync)	I			
	a) Level 2 TDM MUX Common Equipment	As Above + AO–Level 2 TDM FA AO–Level 2 BER Pulses	As Above	
	b) Level 2 TDM DEMUX Common Equipment			
	c) Radio TDM MUX Port			
	d) Radio TDM DEMUX Port			
	e) KG-81 Out-of-Sync			
	f) KG-81 Failure			
(5) Digroup Data Failure (Good Sync)	I			
	a) Level 1 TDM MUX Common Equipment	None	None	
	b) Level 1 TDM DEMUX Common Equipment			

TABLE 7-2 DEB FAILURE/SYNDROME MATRIX (Cont)

FAILURE TYPE	POSSIBLE CAUSES	DOWN-STREAM SYNDROME (OUTPUT SIDE)	SYNDROME	REMARKS
(6) Digroup Data Failure (Bad Sync)	I			
	a) Level 1 TDM MUX Common Equipment	AO–Level 1 TDM FA AO–Level 1 BER Pulses	AO–Level 1 TDM CGA	
	b) Level 1 TDM DEMUX Common Equipment			
	c) Level 2 TDM MUX Port Equipment			
	d) Level 2 TDM DEMUX Port Equipment			
(7) VF Channel Failure	I			
	a) Level 1 TDM MUX Port Equipment	None	None	
	b) Level 1 TDM DEMUX Port Equipment			
(8) Service Channel Failure (Good Sync)	I			
	a) Service Channel MUX Common Equipment	Telemetry Channel Alarm	Telemetry Channel Alarm	Service Channel Outage
	b) Service Channel DEMUX Common Equipment			
(9) Service Channel Failure (Bad Sync)	I			
	a) SC MUX Common Eq. b) SC DEMUX Common Eq. c) Radio TDM MUX Port d) Radio TDM DEMUX Port	Above + SC TDM-FA SC TDM BER Pulses	Above + SC TDM-CGA	Service Channel Outage

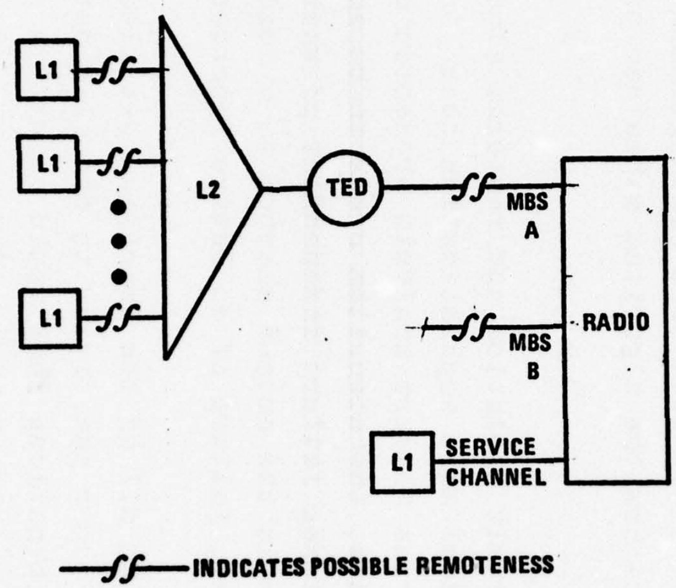


FIGURE 7-1
TYPICAL EQUIPMENT CONFIGURATION

in the receiver where RSL is monitored, it is assumed that the failure causes an RSL alarm. Other transmitter, receiver or modem failures have been assumed to produce an anomolous indication in the signal quality monitor (SQM).

The only finer-grain detail has been to distinguish the cases (which probably have a very low probability) in which a multiplexer common equipment can fail and still produce a good frame sync indication in the corresponding demux or a demultiplexer common equipment can fail without frame alarming. These have been distinguished in order to better illustrate how the algorithm makes use of Restoral Action Lists.

The TSC fault isolation and restoral algorithm can be given a great deal more sophistication than one developed based on the gross failure analysis presented here. For optimum performance, the algorithm needs to be able to recognize a multiplexer failure independent of whether the failure mechanism is the output shorted to a logic level, an open gate input, failure of an entire functional card, etc.

Postulating all of the important detailed failure mechanisms that can occur and trying to manually determine all of the alarm combinations that would result is not a sensible approach to the problem. The recommended method for developing the detailed Restoral Action Lists for an optimized fault isolation algorithm is to field the equipment in a test segment of the network. A special software module should be included enabling the TSC to generate its own detailed failure/syndrome matrix. Then, through a test program that involves the actual introduction of detailed failures, the refined Restoral Action Lists can be generated.

8.0 DEFINITION OF ALARM, MONITOR AND CONTROL REQUIREMENTS

The subject of this section is the transmission equipment alarm, monitor and control points. Considerations discussed include sizing of the data acquisition and control problem, signal characteristics and the problems of scanning for alarm data and addressing of control points. Suggested message formats for data acquisition and control are also presented.

8.1 Catalog of Alarms, Monitors and Controls

Tables 8-1 through 8-4 provide a listing of transmission equipment alarms. These have been generated from the equipment specifications. Included with each alarm is an alarm quality rating and narrative description of the information that the alarm provides.

The definition of alarm quality is based upon the degree of localization provided by that alarm when considered alone, i.e., as though the alarm were the only information available. Combinations of alarms frequently provide a much higher degree of localization. The numeric value associated with the alarm quality is based upon the level of the restoral tree that this alarm allows by itself. The restoral tree and the levels are shown in Figure 8-1. Over the range of the restoral tree, the TSC domain contains five levels which are numbered 0 through 4.

In assigning alarm quality, the quality level is for a failure of the stream associated with the monitor range of the alarm. The alarm quality for any one particular alarm varies substantially as the alarm is applied to stream failures at

TABLE 8-1 ALARM CATALOG - FRC-163

Alarm	Alarm Quality	Information
MBS A/B INPUT D/T	(Stream) 1C	Failure can be caused by KG81 output or radio port hardware.
SCBS INPUT D/T	(Station) 3C	Failure confined to the service channel MUX, radio port hardware or TSC hard ware.
MBS A/B, SCBS OUTPUT D/T	(Station) 3C	Failure can be caused by radio port hardware or attached equipment.
MODULATOR OUTPUT	(Equip) 4C	Failure caused by radio hardware
DEMODULATOR OUTPUT	(Loop) 2C	Failure caused by either transmit side or receive side radio failure
RADIO FRAME	(Loop) 2C	Could be caused by a number of failures in link path.
XMTR FREQ. DRIFT	(Equip.) 4C	Useful as a prodrome of XMTR failure.
POWER SUPPLY	(Equip) 4C	This alarm unequivocally isolates the failure to the equipment.
XMTR POWER	(Equip) 4C	Useful as a prodrome of XMTR failure.
FRAME ERROR THRESHOLD	(Loop) 2C	Potentially useful as a prodrome to false loss of sync declaration.
DIVERSITY SWITCH STATUS	(Equip) 4C	If diversity switch status does not change for some prescribed period of time, a problem with a receiver or the diversity switch is indicated.

TABLE 8-1 ALARM CATALOG - FRC-163 (Cont)

Alarm	Alarm Quality	Information
STATUS (TX1/2, RCVR 1/2, PS 1/2) online/offline	N.A.	Simple online/offline provides no information to fault isolation.
FAILED	(Equip) 4C	Taken in context with online/offline and other radio status, isolates to the equipment level.
RECEIVED SIGNAL LEVEL FRAME ERROR PULSES SIGNAL QUALITY EYE PATTERN	(Loop) 3C	These monitors provide an indication of RF path fade when viewed collectively, these are useful as a prodrome to fade conditions. Taken collectively, these monitors allow a very detailed analysis of fault location within a failed radio.

TABLE 8-2 ALARM CATALOG - TD-1193

Alarm	Alarm Quality	Information
PRIMARY POWER	(Equip.) 4C	Unequivocally isolates cause of MBS outage.
FRAME SYNC LOSS	(Stream) 1C	Flags the occurrence of an MBS outage
LOSS OF OUTPUT (MUX)	(Station) 3C	Unequivocally indicates failed station. Indicates probable cause of MBS outage is level 2 MUX. Problem could, however, be a short circuit on the input line to the walburn.
LOSS OF INPUT (DEMUX)	(Stream) 1C	Indicates MBS outage. Could be caused by most any MBS related equipment along the extents of the stream: i.e., TDM Walburn Walburn Bypass Radio Port
LOSS OF PORT	(Stream) 1C	Indicates the outage of one or more digroups but does not uniquely identify which.
FAULT ALARM	(Equip) 4C	Indicates cause of outage is level 2 MUX. Affected stream could be either digroup or MBS.
FRAME ERROR MONITOR	(Stream) 1C	Potentially useful as a prodrome to false loss of sync declaration.
MUX AND DEMUX ON-LINE MUX	N.A.	These alarms provide no explicit information for fault isolation.
LAST SW. ACTION	N.A.	
OFF-LINE STATE	(Equip) 4C (Stream) 1C	Interpretation of this alarm is dependent upon the state of the online unit. Highly localizing if online unit is go.

TABLE 8-3 ALARM CATALOG - WALBURN/WALBURN BYPASS

Alarm	Alarm Quality	Information
KG 81		
PRIMARY POWER	(Equip) 4C	This alarm localizes the fault to the equipment.
FULL OPERATE	(Stream) 1C	Full operate indicates a resync dialogue is in progress. Persistance of this condition indicates a stream outage.
RESYNC ACHIEVED	(Stream) 1C	Resync achieved indicates a resync dialogue is in progress. Persistance of this condition indicates a stream outage.
SUMMARY ALARM	(Equip) 4C	Localizes problems to transmit side of KG 81
BYPASS STATUS	N.A. C	This status supplies no specific information for fault isolation but is included as part of the critical alarm set because of operational importance.

TABLE 8-4 ALARM CATALOG - TD-1192

ALARM	ALARM QUALITY	INFORMATION
PRIMARY POWER	(Equip) 4C	Unequivocally isolates cause of digroup outage
FRAME SYNC LOSS	(Stream) 1C	Flags the occurrence of a digroup outage
LOSS OF OUTPUT (MUX)	(Station) 3C	Unequivocally indicates failed station. Indicates probable cause of digroup outage is Level 1 MUX. Alarm could, however, be due to a short circuit on the input line to the Level 2 MUX.
LOSS OF INPUT (DEMUX)	(Stream) 1C	Indicates probable cause of digroup outage is Level 2 MUX. Could, however, be due to a short circuit on the input line to the Level 1 MUX or a far-end problem.
LOOPBACK	(Stream) 1C	Indicates unit has been placed in a test mode (taking it out of service).
CGA	(Stream) 1C	Indicates failed digroup (redundant for fault isolation purposes).
FAULT ALARM	(Equip) 4C	Indicates cause of outage is Level 1 MUX. Affected stream assumed to be digroup*
BER PULSES	(Stream) 1C	Potentially useful as a prodrome to false loss of sync declaration.

*Assumption: No monitors on Individual VF channel cards.

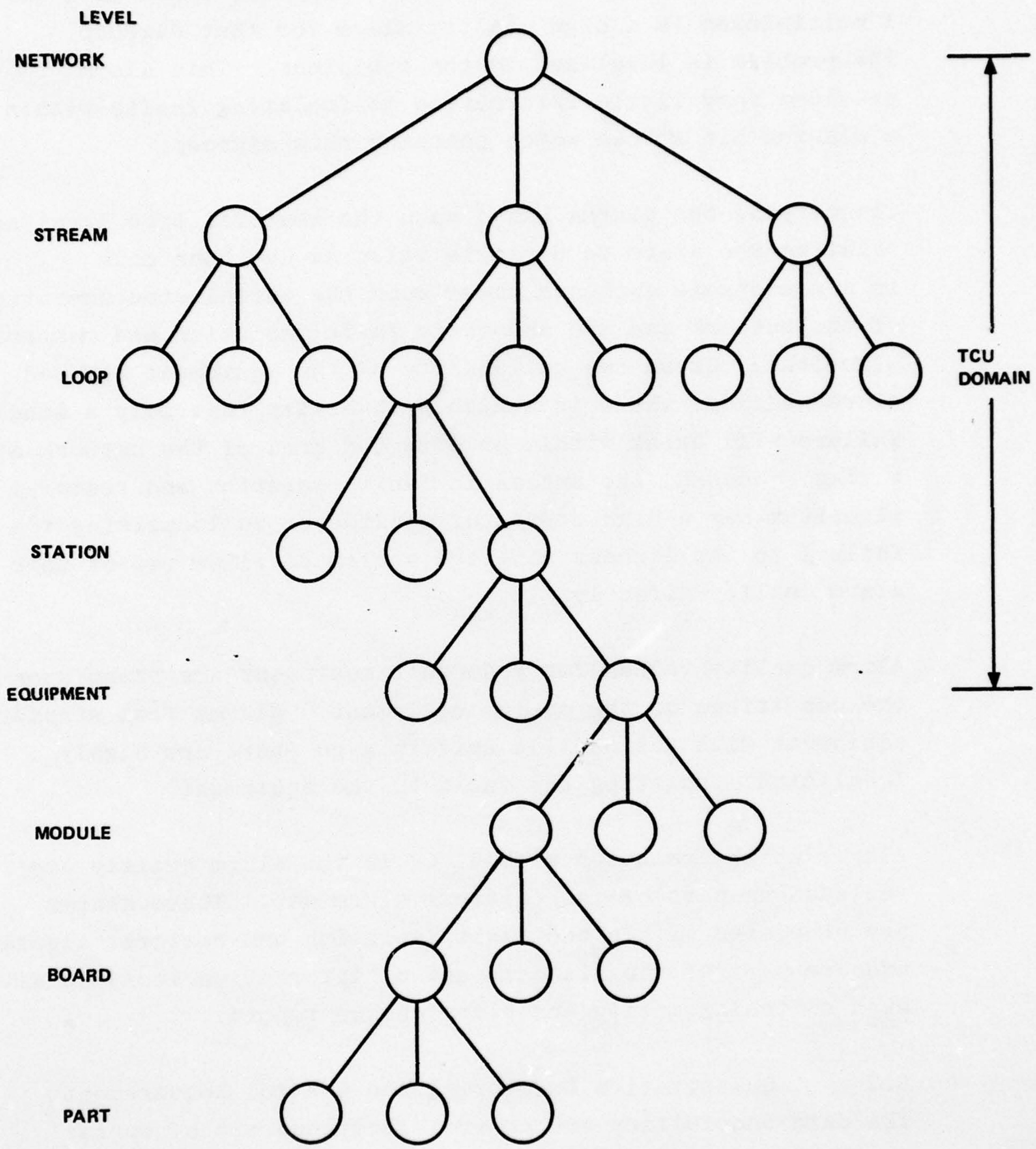


FIGURE 8-1
RESTORAL TREE

higher levels. For example, a primary power alarm in a level 1 multiplexer is a high quality alarm for that digroup. The problem is localized to the equipment. This alarm provides very little information to isolating faults within a mission bit stream which contains this digroup.

Classifying the alarms based upon the restoral tree level and relating the alarm to a single value as has been done is a reasonable approach based upon the anticipated operation of the network and the automatic fault isolation and restoral algorithm. Given the reliability of the equipment and use of redundancy, there is a high probability that only a single failure will exist within an effected area of the network at a time. Second, the automatic fault isolation and restoral algorithm has a high degree of confidence in localizing the failure to the highest affected stream to allow use of this alarm quality directly.

Alarm quality values for redundant equipment are based upon the conditions of the on-line equipment. Alarms from standby equipment with the on-line unit in a go state are highly localizing, isolating the fault to the equipment.

Alarms which are noted with a "C" in the alarm quality are included as part of the critical alarm set. These alarms are processed within the fault isolation and restoral algorithm and are part of the standard set of information included with each switching action and alarm change report.

8.2 Quantitative Monitoring and Control Requirements

The data acquisition and control functions are of course equipment oriented. In order to arrive at a sensible modular partitioning of this hardware, it is necessary to examine the equipment compliments that are encountered at the various stations in the network.

One possibility is to design a branch oriented local data acquisition and control unit. One reason that this is attractive is that the telemetry function is branch oriented and this suggests the possibility of 19-inch rack-mounted Branch Modules where the telemetry and local data acquisition hardware are contained in a single unit. The TSC hardware deployed at each main station would then include one Branch Module for each branch terminating at the station.

Because of the wide variation in the equipment compliment connected to the various branches, the Branch Module approach may not be feasible. It may prove more efficient to configure a general purpose Data Acquisition and Control Unit with one or more of these being required at each station depending on the aggragate of transmission equipment at the station.

In order to arrive at answers regarding the best partitioning of data acquisition and control hardware both at the circuit card level and the unit level, a catalog of the minimum and maximum numbers of monitor and control points has been generated for 1) branches, 2) repeaters and 3) main stations. In doing this, it has been assumed that all the available DRAMA monitor points may be of interest to the tech controller. The monitor and control points for each equipment type are summarized in Table 8-5.

One point of importance is immediately apparent from examination of this table. This is the fact that the Walburn/Walburn Bypass has non-standard alarm and control point interfaces. If the Walburn hardware can be modified to provide TTL level and Form C contact interfaces for all alarm and control points (as does the DRAMA equipment), design of the data acquisition hardware can be greatly simplified.

TABLE 8-5 ALARM, MONITOR AND CONTROL POINT SUMMARY

CONTROLS	TTL LEVEL	TTL LEVEL PULSE	SW CLOSURE
TD-1192	2		
KG-81/Bypass	4	1	1
TD-1193 (Redundant Pair)	5		
FRC-163 (Redundant Pair)	4		

ALARMS	FORM C	TTL	TRANSISTOR SWITCH OUTPUT
TD-1192	8		
KG-81/Bypass	1	3	1
TD-1193 (Redundant Pair)	25		
FRC-163 (Redundant Pair)	30		

MONITORS	PULSES	ANALOG VOLTAGE
TD-1192	1	
KG-81		
TD-1193 (Redundant Pair)	2	
FRC-163 (Redundant Pair)	2	6

A summary of the maximum and minimum branch and station equipment compliments is given below.

- TCU

Branch Minimum (Botley Hill Farm)

1	FRC-163 (redundant pair)
1	KG-81 with bypass
1	TD-1193 (redundant pair)

Branch Maximum (does not occur)

1	FRC-163 (redundant pair)
2	KG-81 with bypass
2	TD-1193 (redundant pair)
16	TD-1192

Station Minimum (Bocksberg)

3	FRC-163 (redundant pairs)
3	KG-81 with bypass
3	TD-1193 (redundant pairs)

Station Maximum (estimated)

12	FRC-163 (redundant pairs)
20	KG-81 with bypass
20	TD-1193 (redundant pairs)
80	TD-1192

- RTU

Station Minimum (unmanned repeater)

2	FRC-163 (redundant pairs)

Station Maximum (Hann)

2	FRC-163 (redundant pairs)
4	KG-81 with bypass
4	TD-1193 (redundant pairs)
5	TD-1192

TABLE 8-6 QUANTITATIVE ALARM AND MONITOR SUMMARY

	BRANCH			RTU STATION			TCU STATION	
	MIN.	MAX.		MIN.	MAX.		MIN.	MAX.
CONTROLS	15	58		8	62		45	428
ALARMS	60	218		60	220		180	1600
MONITORS (Pulses)	4	22		4	17		12	144
MONITORS (Analog)	6	6		12	12		18	72

The Quantative Alarm and Monitor Summary presented in Table 8-6 was generated from the above plus the data given in Table 8-5. Here is has been assumed that the Walburn controls and alarms have all been converted to TTL level and Form C respectively.

8.3 Response and Accuracy Requirements

8.3.1 Alarm Change Response Time

Alarm information as a result of equipment state changes is required by both the operations personnel and by the automatic fault isolation and service restoral algorithm. Comparatively loose requirements are appropriate for information destined to the operators. Times on the order of 1 or 2 seconds are appropriate since they are within the same range as human response time. One complicating factor is the fact that the DRAMA equipment usually produces multiple alarm changes with any given fault. A reasonable requirement for operator display information is an average delay of 1 sec or less with 90 to 95% of all infomration to be displayed in 3 sec or less.

Information to be passed to the automatic fault isolation and service restoral algorithm requires much more stringent time delay requirements. Performance of the algorithm is dependent in a large part on DRAMA equipment resynchronization time and propagation of data acquisition information. In order to not contribute appreciably to the delay experienced in determining the success or failure of an attempted restoral action, the response time should be kept below about 15 msec.

8.3.2 Control Response Time

Control response time is divided into two phases. The control function must pass from the originating TCU to the destination TSC processor. Once at the destination TSC processor, the control command must be interpreted and executed.

Overall response to control commands is slightly less demanding than response to alarm changes because of the relative frequency of the two and their respective contribution to the overall system performance.

Since control of the DRAMA equipment is comparatively simple given a request for some control action, the second phase (TSC processor to DRAMA equipment) can be made very rapid if some care is exercised in the hardware/software design. With this assumption, the first phase becomes the controlling factor.

Operator generated control commands can be initiated from remote locations and may required interloop routing. The interloop routing delay experienced at each node will contribute to the overall delay. The limiting factor in this case should be based upon the human involved which would again place an average delay of 1 sec or less with 90 to 95% of all operator initiated control functions occurring 3 sec or less.

Control functions as part of the automatic fault isolation and restoral algorithm are limited to the domain of the local loop and no routing is required. To operate at a level that allows equipment resynchronization to be a limiting factor requires control action and response time on the order of 10 to 15 msec. Some additional time is involved within the automatic fault isolation and restoral algorithm which would allow this action and response time to be increased to 25 msec.

8.3.3 Dynamic Range and Accuracy of Analog Measurements

Purely analog signals within the DRAMA equipment are limited to power supplies and signals associated with the FR-163 radio set (transmitter power, received signal level, and signal quality monitor). Included also in this general area are the pulses associated with frame BER since the result of these pulses can be associated with a continuously variable function. Analog measurement of power supplies is not

considered since primary power alarms are specified and operational limit alarms are assumed.

There are essentially two uses for the information contained within the analog signals. First, there is a need to supply the operations personnel with numeric values that can be used in determining the performance of the equipment. Second, the analog signals can be used to derive information for automatic fault isolation and restoral algorithm.

For the purposes of the automatic fault isolation and restoral algorithm, the analog signal information is useful only when it passes through a level which determines a fault condition. The algorithm, as it is described, makes no use of a numeric value per se. Operations personnel will require a numeric value that is of adequate accuracy to assess station performance and will also require that the information be presented in meaningful units of measure.

One consideration with regard to analog signal measurement is the frequency spectrum of the signal. Since measurement of the absolute value of each of the signals is essential, the domain extends to D. C. at one end. As the bandwidth of the data acquisition circuitry is extended from this point, the time response to analog signal change is improved. However, as this time response is improved, the number of potential changes that are of importance increase. These can impact the overall system adversely by increasing the load placed upon the station processing capacity and local loop telemetry channel activity.

Limiting analog frequency response to something on the order of the other alarm changes (i.e., 50 msec) also limits the utilization of the analog signals as prodromes (trending). While such utilization is not discussed in any detail within this report, TSC action based upon prodromes of these signals is likely feasible and may assist in reducing declarations of loss of sync and subsequent frame sync searching during short term outages such as those caused by fades.

A reasonable compromise for the time resolution provided in measuring analog signals would be on the order of 1/4 to 1/3 of the resync time of the DRAMA equipment. This yields a frequency response on the order to 80 Hz if full-scale signal swings are assumed.

The dynamic range of received signal level is taken as 55 dB based on the FR-163 radio set specifications. A similar dynamic range can be assumed for the signal quality monitor. Resolution and measurement accuracy of these signals to levels smaller than .5 dB is not warranted for operator display. Measurement of frame BER pulses extends from .5 to 0 BER. Frame BER in excess of 1×10^{-2} approaches the level of loss of sync. Frame BER less than 1×10^{-7} produces very few pulses, even at the 12 Mbps rate found at the radio. This implies an operations personnel useful range from 1×10^{-1} to 1×10^{-6} with indications that BER lies outside of this range on both ends.

8.4 Data Acquisition

There are a number of factors that bear on the choice of a reporting scheme. These considerations include: minimization of telemetry channel utilization; minimization of the processing load placed and the RTU and TCU; minimization of the TSC response time to alarm changes; and information reliability.

8.4.1 Local Loop Reporting Options

Three alternatives were considered for local loop data acquisition scanning. These included, 1) reporting of all

raw data to the cognizant TCU on a routine basis, 2) periodic poll/response reporting of a subset of critical alarms, and 3) reporting data only when an alarm change occurs.

The transmission of all raw data associated with a local loop every frame is impractical for two reasons. First, the volume of data that must be sent implies a long frame period. A long frame period will result in a poor response time to an alarm change and will increase the interloop routing delay. Secondly, transmission of all raw data places an unnecessary and unacceptable load on the TCU processor.

Transmission of critical alarm data on a routine basis was rejected for basically the same reasons: changes will be very infrequent and it is senseless to load the channel with a lot of redundant data when it is not needed.

The recommended reporting method is a report by exception scheme. Alarm points are scanned by the TSC data acquisition hardware and comparison is made with the results of the previous scan. If nothing has changed, no data is sent. In this normal mode of operation, an Optional Response Poll is periodically transmitted by the primary station and circulated around the loop. Since the normal condition produces no responses, the delay between polls is equal to the loop propagation delay. Scanning of loop stations is thus very rapid. For reliability purposes, the loop primary may periodically send a Required Response Poll to which each secondary must respond with an "A-OK" message. Simply observing the return of the Optional Response Polls, however, give a fair degree of confidence that the telemetry subsystem is functioning properly.

In addition to providing rapid scanning of local loop alarms, the report-by-exception method minimizes inter-loop routing delay and TCU processor loading.

8.4.2 Stream Status Reporting

It is important to note that the TSC data acquisition problem extends beyond the local loop. Remote data stream associated alarms represent key information for fault isolation. This remote alarm information is required by all TCU processors along the route of the stream.

In order to distribute remote alarm information, stream status messages were devised. These messages are routed by a special handler so that they pass along the exact route of the associated stream. These messages are received and processed by each TCU along this route.

Information contained within the stream status message identifies the stream, provides the failed/not failed, explained/not explained status, and the goodness (with respect to the degree of fault localization) of the alarm information. These messages are discussed in greater detail in the sections of the report dealing with the automatic fault isolation and restoral algorithm.

8.5 Formats

8.5.1 Data Acquisition Formats

There are 3 basic types of data acquisition activity within the system, each requiring some special (or different) considerations and processing. First, there is routine local loop data acquisition; second, requests for raw data; third,

AD-A049 225 E-SYSTEMS INC ST PETERSBURG FL ECI DIV F/G 17/2.1

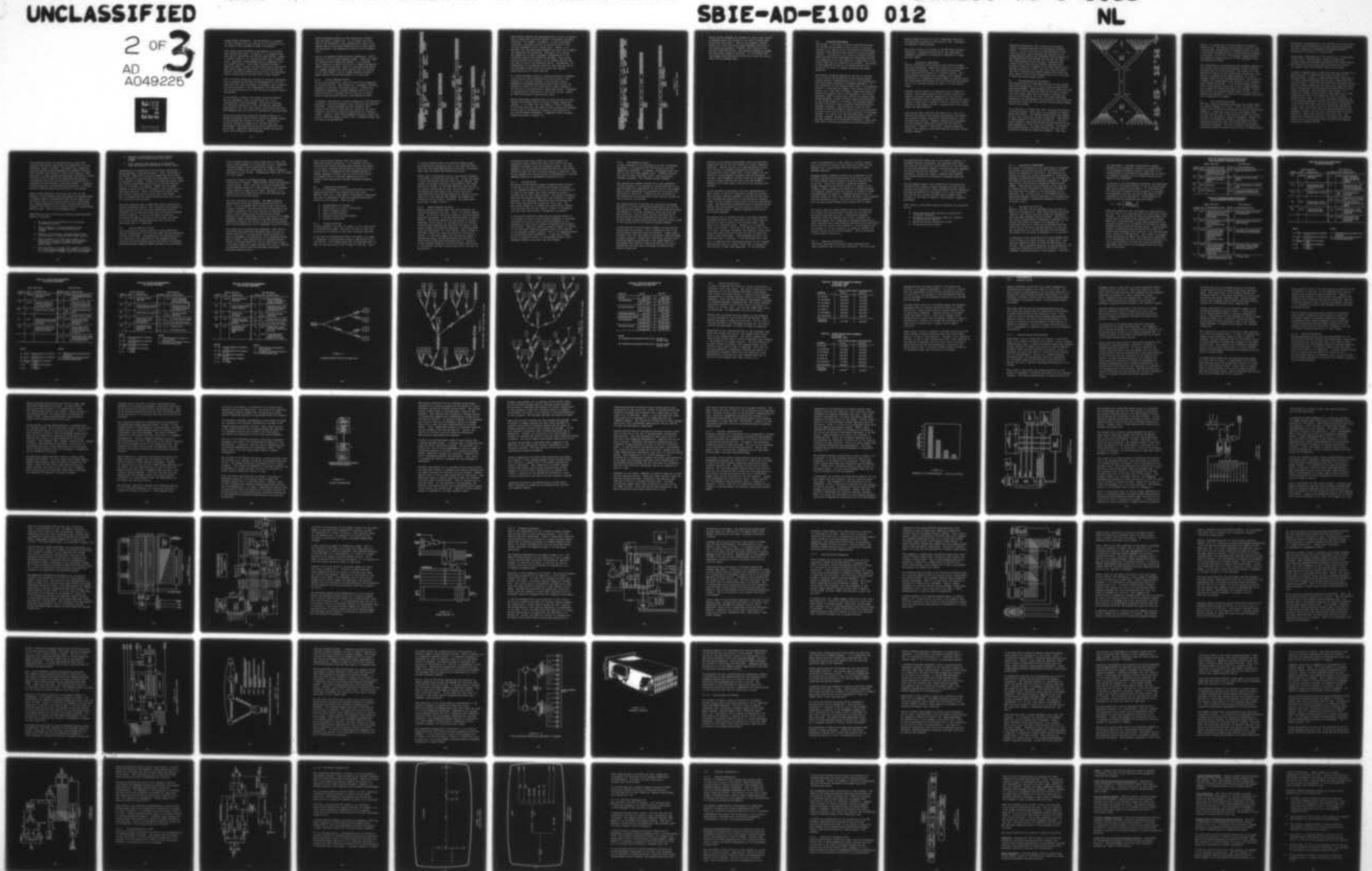
TRANSMISSION SUBSYSTEM CONTROL ANALYSIS AND DEVELOPMENT.(U)

JUL 77 R K SMITH, J N BEAUCHAMP DCA100-76-C-0036

UNCLASSIFIED SBIE-AD-E100 012 NL

2 OF 3

AD A049225



OF 3

9225

stream status information. The differences in treatment of these streams are in the end use of the information and the range of this information within the network.

Both stream status information and requests for raw data have network wide range. In general, it is expected that this type of information can and will pass through a number of control loops as it is directed from origin to destination. Therefore, these messages must be compatible with any network protocol. The primary difference between these two forms of traffic is that stream status information must travel along the exact path of the stream within the network. Raw data information has an end to end requirement with no restrictions on the path that it must traverse.

Local loop data acquisition contains the detailed information that is required by the TCUs for determination of the loop state and local loop involvement in automatic fault isolation and service restoral. As a consequence, this information will be contained within the loop and has no network protocol requirements.

In the design of formats, the overall system performance must be considered. This is especially true in the considerations of TCU processing. Along with containing all functions required for an RTU, the TCU has additional responsibility for local loop control and network activities. This suggests that the formats be optimized for TCU processing.

TCU activity with data acquisition consists primarily of message processing, error control, and use of message content if the contents are relevent to this TCU. Ignoring the potential increase in memory size, the most efficient processing is characterized by very direct paths with very few actions. Decisions concerning the formats can greatly effect the efficiency of TCU processing.

Detailed message formats for data acquisition information are shown in Figure 8-2. In all cases, the first byte of the information field specifies the class of the message which allows rapid identification so that control may be passed to the appropriate function with a minimum delay.

Most of the message formats are self-explanatory. An exception is the no change response format. Under the report by exception data acquisition scheme, it is expected that a majority of polls will elicit no change information. Within the link protocol, there is a polling format which is a demand resonse poll which requires that all devices on the loop respond. The no change response provides a simple check on the TSC functioning in a local loop.

8.5.2 Control Message Formats

As with data acquisition, there are three broad catagories of control activity. These areas are: equipment switching requests for raw data; and system control. Unlike data acquistion information, the most general case of all of these activities is network wide and thus all of these control functions must conform to the network protocol.

In general, the sources of control message are from operations personnel or from the automatic fault isolation and restoral algorithm. Control messages from the automatic fault isolation and restoral algorithm are limited to the local telemetry loop and are potential candidates for a separate format. However, since these messages eventually access the control functions within an RTU or TCU, a separate format does not seem to be appropriate.

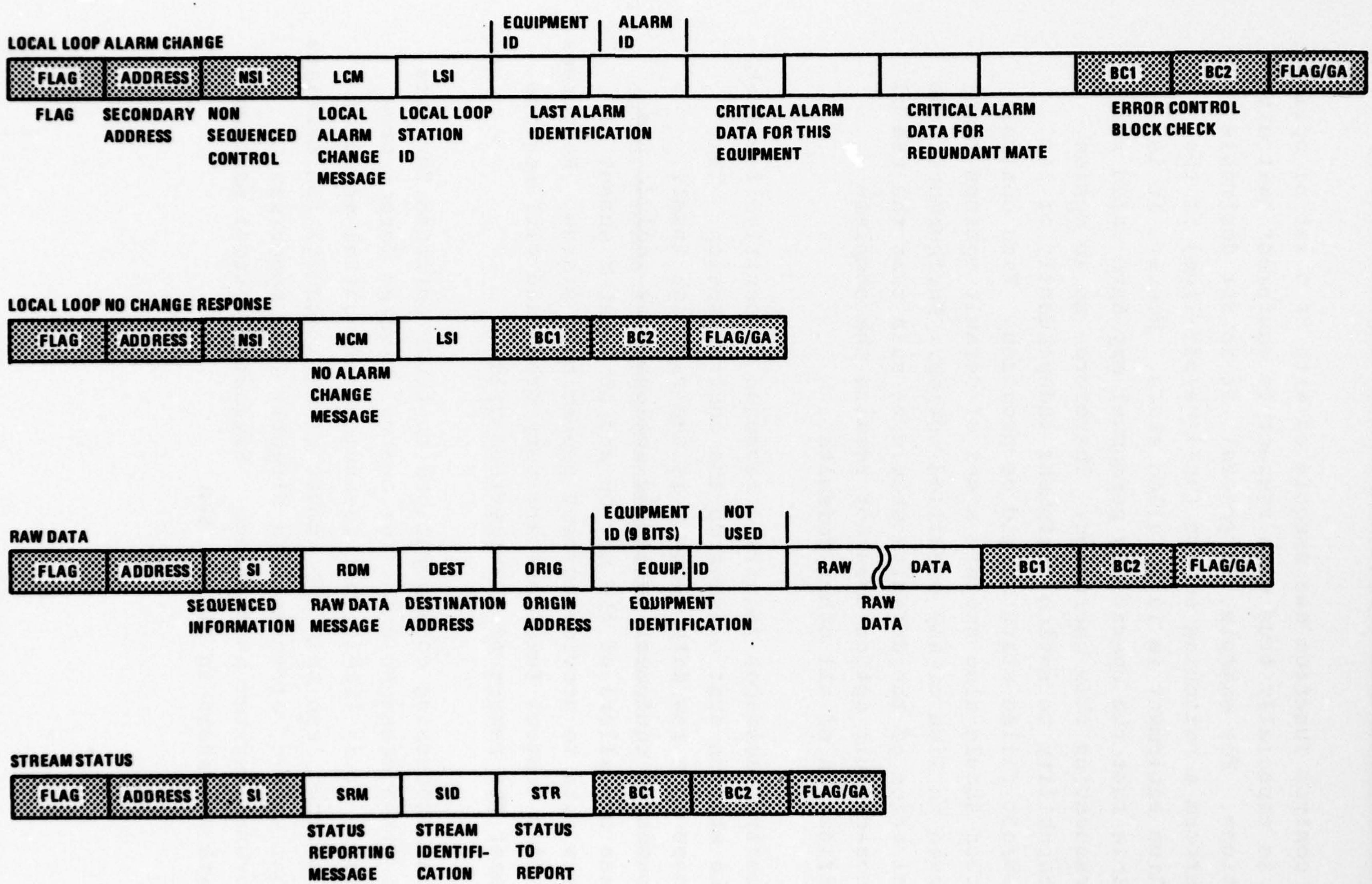


FIGURE 8-2
DATA ACQUISITION FORMATS

Each control function has associated with it a set of options. This is especially true with respect to equipment switching functions. For example, in general, it is not desirable to perform a switching action (on-line/off-line) if the off-line equipment is in a failed state. Hoever, it is possible that the operations personnel may have valid reasons for requesting this operation. Therefore, as an option, the capability to swtich equipment independently of its failed/not failed state should be provided. Each control function should also contain a set of default options that are used in lieu of any specified option. Furthermore, the organization of the defaults should be such that the use of one non-default option does not require the complete specification of all other defaults.

All control messages require a response as positive feedback to the station that originated the control action. The response to a raw data request is the raw data itself. Responses to equipment switching actions must include the success or failure of the action and at least a subset of the raw data to provide current operational status. Responses to system control functions are very broad and will require at least the return of the previous value.

Similar processing considerations to those outlined for data acquisition messages exist for control message format processing. It is likely that the number of control messages will be much less than the number of data acquisition messages so that loss of efficiency is slightly less important to the overall system performance. Suggested control message formats are shown in Figure 8-3.

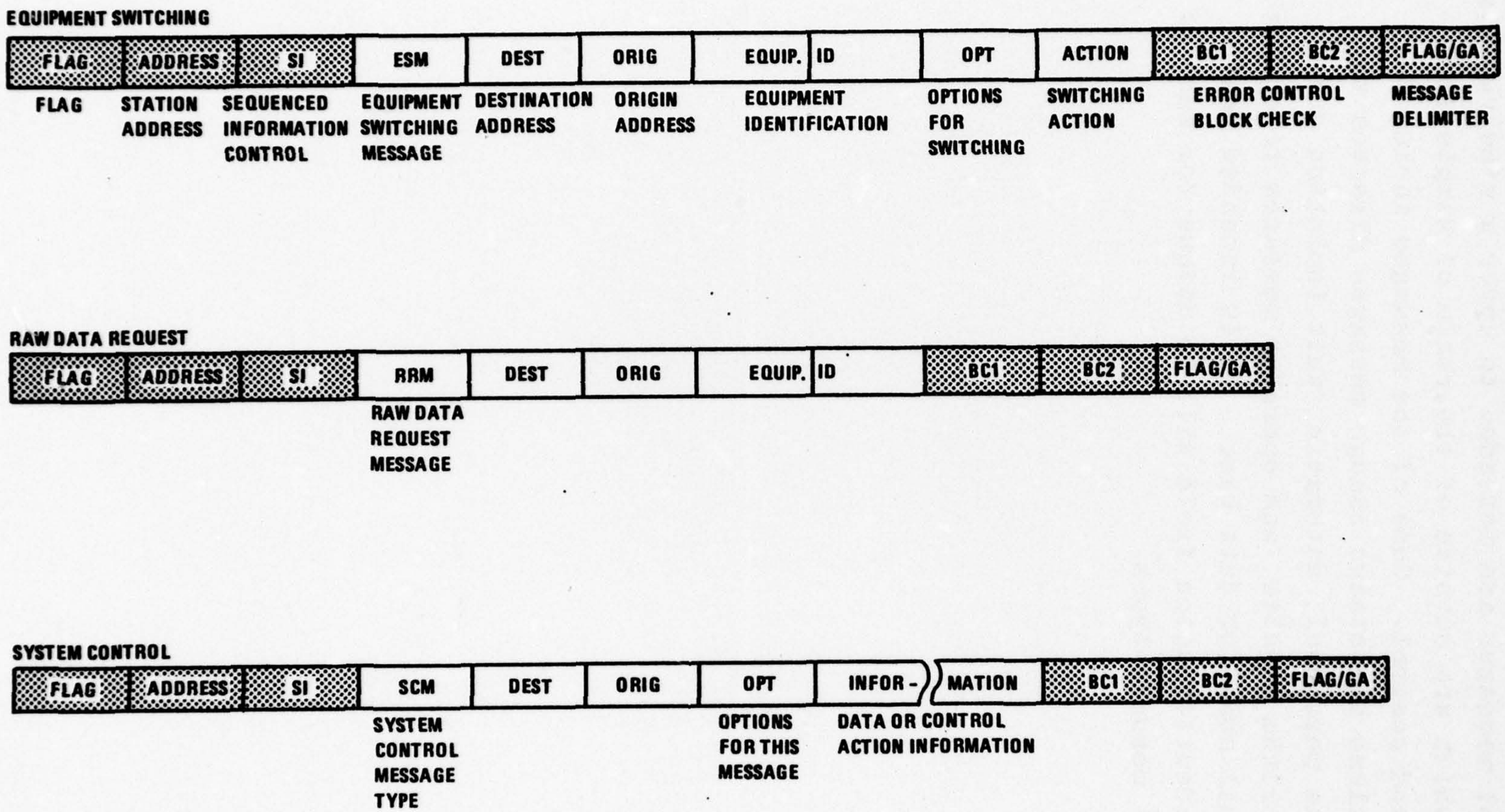


FIGURE 8-3
CONTROL MESSAGE FORMATS

System control messages are intended to cover a wide variety of messages which are outside of the realm of simple data acquisition and control. Some of the messages included within this class are status change messages directed to the operations personnel, automatic fault isolation and restoral algorithm results, and operator messages to change TSC operational modes or data base. It is intended that the message identification field will be unique for each of these control message types.

9.0 ALGORITHM DEVELOPMENT

The purpose of the fault isolation algorithm developed for the TSC and discussed in this section is two-fold. One goal is to improve circuit availability by performing automatic fault isolation and restoral in place of manual restoral. A second goal is to reduce tech controller manpower and skill-level requirements by diagnosing failures and presenting to the tech controller a description of the failure mode that is a definitive as possible. The purpose of the algorithm is thus not to replace manual fault isolation but rather to augment and assist the manual operation.

One powerful technique employed by the TSC in isolating faults involves the correlation of data stream alarms. Another powerful troubleshooting tool is the systematically controlled remote switching of redundant equipments. By the correlation of data stream alarms alone, it is generally possible to determine the highest level stream failure that exists, e.g., link, MBS or digroup. Unless the failure is explicitly alarmed by the failed equipment or the failed equipment causes an unequivocal alarm in an adjacent down-stream equipment, further isolation beyond the declaration of the stream outage is, in general, not possible unless the TSC is given the power to perform remote switching of redundant equipment. This switching is thus not only required to provide automatic restoral but is also essential in order to do a decent job of fault isolation to aid manual restoral. Of course, once this switching power is given to the TSC, automatic restoral is automatically provided, provided restoral can be achieved by switching. Thus, automatic fault isolation and restoral and fault isolation to

support manual control are really inseparable functions. The algorithm discussed in this section, then, represents an approach to both problems.

This section given an overview of the TSC fault isolation algorithm. A detailed description of the algorithm is presented in Appendix A along with an analysis of outage restoral times.

9.1 Alternative Approaches

Several alternative fault isolation and restoral schemes have been considered. These vary primarily in the degree of distributed processing. The first method centralizes fault isolation and restoral at a single location. The second method distributes fault isolation and restoral processing to a comparatively small number of regional sites. The third method distributes the fault isolation and restoral over all of the main stations or over all of the control loops.

Complete central control is unattractive for a number of reasons. This scheme requires that a single central site maintain the entire network connectivity, all critical alarm information must be forwarded to the central site which implies a large flow of telemetry traffic, and since control resides in this single location, failures within this station impact the entire network.

Distributing fault isolation and restoral to a set of regional sites improves the overall survivability and reduces the telemetry traffic required to perform automatic fault isolation and restoral. However, connectivity information for each digroup, mission bit stream and link must be maintained. Further, a need exists to pass information between these regional sites since digroups can pass through multiple regions.

Completely distribution fault isolation and restoral results in maximum survivability and minimum telemetry traffic. If fault isolation is distributed down to the local loop level, fault isolation and restoral can be accomplished by knowing the failed/not failed condition of the streams which pass through the loop and the alarm data from equipment within the loop. Completely distributed fault isolation and restoral requires that a decision be made as to whether or not fault isolation and restoral should occur within this local loop and, given that fault isolation and restoral should occur within the loop, which of the two main stations which are at the ends of this loop should assume control.

9.2 Stream Designation and Extents

The mechanism exists to determine the state of the streams which pass through the local control loops through status reporting messages. These messages which are routed over the exact path of the associated stream can be used in conjunction with simple, small data structures to maintain the state of the streams of the loop. With some intelligent inferences, the determination of the need for fault isolation and restoral within the loop and which station should assume control can be made.

It is important to realize that fault isolation and restoral is stream oriented. Also, there is a hierarchy of streams within the network. The defined streams within the network are the link, mission bit stream and digroups. The extents of these streams is illustrated in Figure 9-1. The link exists between 2 radios and includes the RF path and all the radio hardware up to the radio port hardware. The mission bit stream maintains its unique identity from the level 2 multiplexer through the radio port hardware. After this

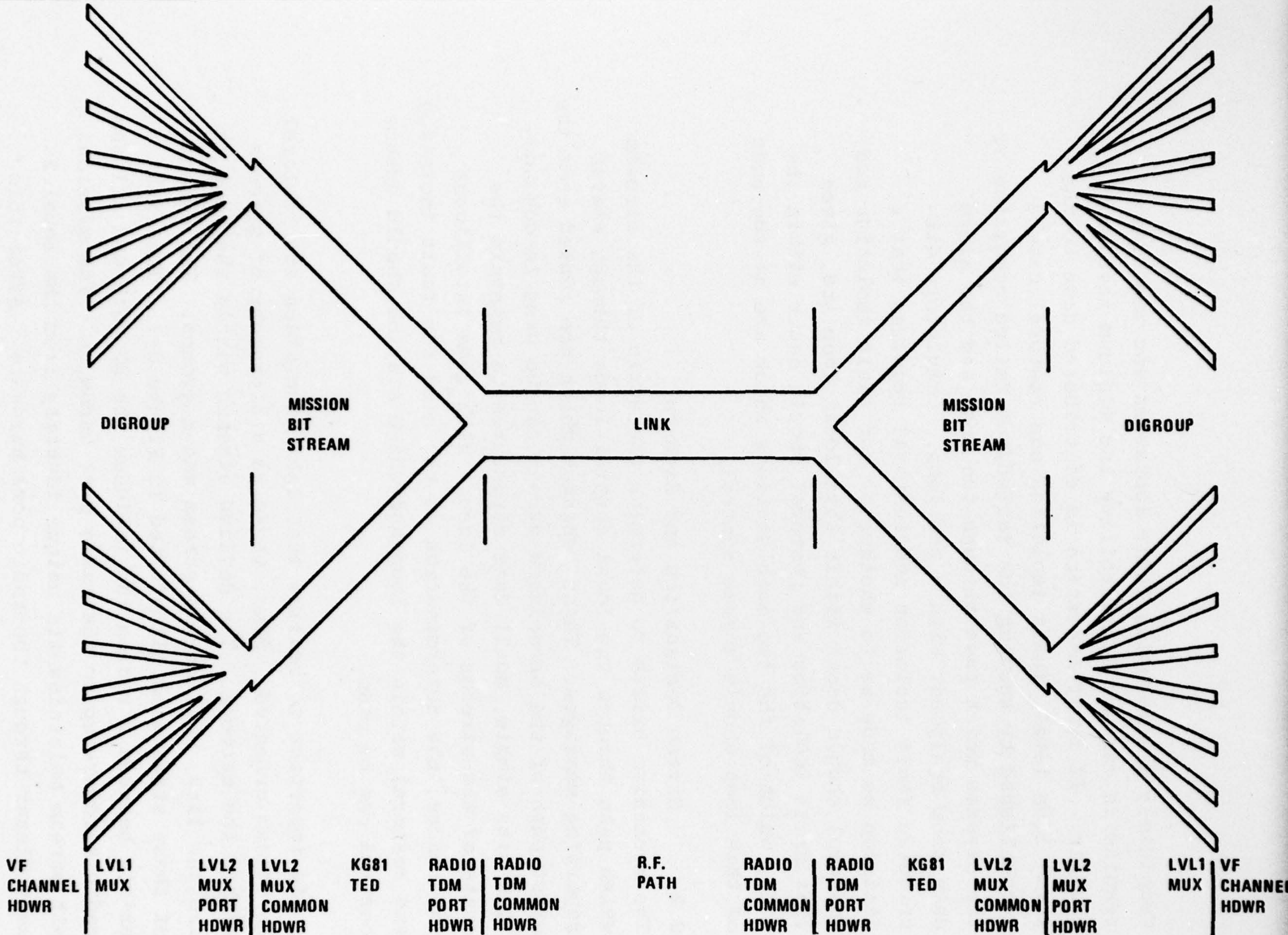


FIGURE 9-1
DATA STREAM EXTENTS

point, it is merged with a second mission bit stream to form the link. The digroup exists between a pair of level 1 multiplexers and maintains its identity within the level 1 multiplexer through the level 2 multiplexer port hardware. At the point of the level 2 multiplexer common hardware, the digroup is merged with up to 7 other digroups to form the mission bit stream.

This observation of the definitions and equipment extents of the streams within the network suggests a fault isolation procedure that is stream oriented. The failure of a link implies the failure of the 2 mission bit streams and a link failure is confined to the common equipment of a pair of adjacent radios and the intervening RF path. The failure of a mission bit stream implies the failure of the 8 digroups which compose the stream. It is confined between a pair of level 2 multiplexers at the level of their common hardware and includes the 2 KG-81s and radio port hardware. The digroup failure is confined between a pair of level 1 multiplexers and includes all of the level 2 multiplexer port hardware through which it passes.

9.3 Fault Isolation Overview

The fault isolation procedure based upon the streams within the network operates at 2 levels. First, the failure of a stream can be inferred by the failure of all the streams which compose it. Given that there are no other higher level failures that are part of the stream or include the stream, the fault is isolated to a group of equipment. Second, the failure of a stream can be directly determined by alarm conditions within the local loop as detected by the data acquisition equipment. The first condition corresponds to unalarmed equipment fialures and the second to alarmed faults.

Having made a determination of the failure of a stream within the local loop as detected by the data acquisition equipment. The first condition corresponds to unalarmed equipment failures and the second to alarmed faults.

Having made a determination of the failure of a stream within a local loop, a decision must be made to begin fault isolation among the equipment specified by the stream failure. The decision is solely based upon the presence or absence of a higher level failure which contains the stream.

A number of methods exist to make this decision including telemetry inquiries to other stations along the path of the failing stream. However, a simple method exists with the use of status reporting messages and the use of some additional data structures. Along with maintaining the failed/not failed status of each stream within the local loop, additional information representing the explained/unexplained status of each stream is required. The procedure is as follows. If a mission bit stream failure occurs at some remote site, frame alarms and bit error rate alarms will occur at certain affected local loops. At a local loop which terminates any of the affected digroups, the TCU will declare these digroups to be failed based upon local data acquisition information. If this local loop defers action on the failed digroups for a short time after it has sent the required digroup failure messages, a report from the local loop containing the failed mission bit stream will arrive at this local loop. The mission bit stream failure report will explain the digroup failure and will thus inhibit fault isolation and restoral based upon a digroup failure.

This declaration-wait for explanation will assure fault isolation and restoral based upon the highest level failure affecting a group of streams. As higher level streams are restored, failure restoral messages are required to unblock fault isolation and restoral on lower level failures. From this point, a simple decision process must be derived to assure that the action is taken by the appropriate main station. It should be noted that failures within the network can be classified as receive-side or send-side failures. In general, the most sensitive and sometimes only indication of a stream failure is derived from the receive side of that stream.

Based on this observation and noting also that the vast majority of the failures within the network are unidirectional, it becomes logical to designate the main station which is on the receive side of the failing stream as the station which must assume responsibility for fault isolation and restoral. A special case exists in the situation in which a bidirectional failure exists. This will be discussed later.

Summarizing the fault isolation and restoral algorithm development to this point:

- The algorithm is distributed over the network to each main station.
- Fault isolation is stream oriented to link streams, mission bit streams and digroup streams.
- Restoral is hierarchial, restoring higher level failures prior to restoring lower level failures.
- Fault isolation of a lower level stream is inhibited by a failure of a higher level stream by status report messages reporting the higher level failure.
- Fault isolation of a lower level stream is enabled by a status report message of a higher level restoral or by the absence of higher level failure messages.

- Restoral is controlled by the main station closest to the receive side of the failed stream.

- Fault isolation and restoral is initiated by inferred failures as determined by status report

Determination of stream status from the data acquisition system generates 3 possible outcomes. First, the stream is not failed. There are no critical alarms from the on-line unit, the state of the off-line unit determines if the stream is vulnerable to outage. Second, the stream has failed, there are critical alrms from the on-line unit and the off-line unit (if it exists) is alarming. These critical alarms, however, could be result of a failure at this location or at some other location. Third, the stream has failed, there are critical alrms form the on-line unit; the standby unit is alarming and the critical alarms can only be a result of a failure at this location.

This last condition adds an additional fault isolation and restoral procedure inhibit. Fault isolation and restoral in this situation will not cure this problem and should be inhibited for the affected stream until the faulty equipment has been reparied. As far as the impact on the data structures throughout the path of this stream, the result is a status report message indicating the failure and indicating that it is explained.

9.4 Service Restoral Action

Action required for service restoral follows immediately from the determination of the failure mode and the associated restoral action list. The ordering of the switching action is a function of the likelihood of the equipment having failed an how that swtiching action will affect the streams. Switching between redundant DRAMA equipment can be performed

with no adverse effect as long as some care is taken that known failed equipment is not switched on-line. Resyncing the KG-81 causes a short bidirectional outage of the associated mission bit stream which will cause a temporary disruption of data traffic. Bypassing the KG-81 must be viewed as a last resort.

During the course of switching equipment, some switching actions will be blocked. These blocked actions are generated from 2 sources. First, both the on-line and standby unit can be alarming because of a fault outside of the equipment itself such as a common loss of input. Secondly, it is possible that the stand-by unit has failed previously and the on-line unit has now failed.

The first case poses no problems. The DRAMA equipment specifications state the automatic switch-over will not occur. If it is assumed that a similar action occurs with a remote switchover, the switching action requrested by the telemetry system will not occur and will be indicated by a lack of status change in the on-line/off-line status. Assuming that this inhibiting action is not part of the DRAMA equipment, simple interrogation of the data acquisition will immediately indicate the failure of the on-line and standby units and switching action can be suppressed by the TCU itself.

The second case is slightly more demanding. This case requires that unalarmed failed equipment must be stored by the TCU and prior to initiating any siwtching action on a piece of equipment, the possibility of an unalarmed failure must be tested. This requires that an unalarmed failure status be included within the critical alarm areas as part of the data acquisition module and also that service restoral via equipment declare switched equipment failed when switching restores service.

Fault isolation and restoral must then provide some additional system information. At a minimum, the fault isolation and restoral procedure must provide the results of its action (restored service/did not restore service), actions which it attempted but were blocked because of standby equipment failures, and the location and status of equipment which it switched immediately prior to restoral which potentially contains unalarmed equipment faults.

9.5 Exceptional Conditions

So far, the development of the algorithm handles the well-organized portions of the network where they exist. However, there are a large number of exceptional conditions that must be addressed to make the algorithm operational within the real network. These include:

- unused mission bit streams
- unused ports in a level 2 multiplexer
- service channel failure
- power up initialization
- failures within the TSC hardware
- operator intervention

9.5.1 Unused Facilities

In many instances, one port of a radio or one or more ports of a TD-1193 are unused. The effect of this is that less information is available to the fault isolation algorithm.

In the case of an unused radio port, the effect is to make it impossible to discriminate between an MBS failure and a link failure. This simply means that the algorithm must act upon two restoral action lists instead of only one. This

is easily handled without any algorithm changes simply by executing the appropriate part of the algorithm twice - once with the unused MBS declared failed and, if necessary, again with the unused MBS declared not failed.

Unused level 2 multiplexer ports or non-existant digroups are handled in essentially the same way. In the case where a level 2 multiplexer contains only a single digroup, this reduces to the same situation as discussed for the unused mission bit stream. In the case where more than one port of a level 2 multiplexer is used, the following occurs. If the unused ports are taken as not failed, the alarm correlation can never declare mission bit stream as failed since all of the digroups that compose the mission bit stream will not be failed. Therefore, the unused input ports must be taken as filed.

9.5.2 Service Channel Failures

Service channel failures impact the fault isolation and restoral procedure in two ways. First, the service channel serves as the data acquisition medium to this distributed algorithm and second, it allows remote swtiching to restore service. The service channel has its own set of failure syndromes, some of which are included as part of the data stream failure syndromes and some which are unique to the service channel and telemetry subsystem. Conditions which cause the declaration of a link failure, either alarmed or unalarmed, potentially imply the failure of the service channel. It has been assumed that the service channel is implemented using a TD-1192 multiplexer and thus has a frame alarm and BER alarm associated with it. Peculiar to the telemetry subsystem is the block check code as part of the protocol and loss of polling activity, also part of the protocol.

Correctable faults that affect the service channel are contained entirely within the radio since this is the only redundant equipment within the telemetry system. The radio equipment will be switched by whichever TCU still has control capability when the loss of service channel is detected. This action together with service channel reconfiguration to provide degraded operation is discussed in Section 6.2.2.2.

9.5.3 Initialization

The status of the various streams flowing through a station is the key to the fault isolation and restoral procedure. If the telemetry processor is out of service for some period of time for any reason, the status information which it currently has is very likely invalid. The status information which can be obtained through local loop telemetry is easily gathered. The remaining information must be obtained from the stream ends which are outside the local loop. The most direct method is to request the stream status information from the loop far end, however, there is no assurance that the other main station's information is accurate.

A more positive method of obtaining the required status is to initiate a series of status request messages to the stream ends which will elicit status reporting messages to the equipment far ends just as though an equipment status change had occurred. Since the overall telemetry channel utilization is low and status reporting messages require minor amounts of the telemetry channel resource, this is the suggested method of reacquiring status information. It may also be appropriate to reacquire status on a prescheduled basis, after equipment repairs, and after long fade outages.

9.5.4 TSC Hardware Failures

To take advantage of the survivability of this distributed algorithm, it is important that the failure of the TSC hardware be considered. In general, the hardware must fail soft or fail into some degraded state. Hardware failures can occur in three functional areas: data acquisition hardware, processor, and telemetry channel hardware.

Data acquisition hardware failures will cause impossible failure syndromes to be generated or alarm conditions to be declared which are inappropriate to the state of the network. Some of these conditions can be detected at the station and acted upon at that point. Others will not be obvious to the station but can be detected by other stations as part of fault isolation and restoral.

The simplest method of handling data acquisition equipment failures is to allow some forcing conditions as part of the data acquisition module. These forcing conditions would allow the masking out of faulty information until the fault had been corrected. Some of these forcing conditions can be generated within the TSC algorithms but the safest way is to enter these forcing conditions under operator control either at the failing site or via telemetry channel command.

Certain sections of the processor hardware can be monitored continuously for failures. This is especially true of any and all memory associated with the processor. Simple parity provides a very low cost method of detecting a high percentage of memory failures. Single error correcting, multiple error detecting hamming codes provide a much higher degree of performance with respect to memory reliability and can be implemented with only small percentage increases in the overall system cost.

Failures in and around the processor itself are much more complicated and are not well handled by current commercial practices. Typical practice in a high reliability system is to utilize a redundant configuration which greatly increases the overall system cost. A more common low cost method is to use diagnostic software and perform margin testing. Good diagnostic software tends to be comparatively large and would increase the cost of the TSC hardware because of the extra memory required if such software were resident.

In the interest of providing a low cost implementation, a small subset of the total diagnostics should be included with the basic software resident at each RTU and TCU along with providing the capability to load diagnostic software either via the telemetry system and/or by on-site maintenance teams. The abbreviated diagnostics would be executed during periods when the processor is not actively involved in any mission oriented tasks. Complete diagnostics can be run on a preset schedule.

One feature can be included as part of the basic hardware configuration which is a very useful fault indicating tool and can be used to assist in the fail-soft mode. This feature is a watch-dog timer. Operation of a watch-dog timer is as follows. Initially, the watch-dog time is set to a value. Scattered throughout the program at major entry points are small sections of code that set the watch-dog timer to that value again. During program execution, the watch-dog timer is running. If the watch-dog timer times out, it indicates that the program section did not complete execution and there is likely a fault that is preventing that section of code from completing. If the watch-dog

timer is implemented as a logic signal or contact closure, this can be used to control lines which will force the TSC hardware into a failed state which will permit at least degraded operation.

Telemetry channel hardware failures can be detected by the processor through the various status lines which are part of the telemetry channel hardware along with detected loss of polling. Within the station, a detected telemetry channel hardware failure should result in a bypass of the telemetry channel hardware and effectively through-grouping the telemetry channel within the station to achieve at least a degraded mode of operation. Bypassed operation will be detected within the local loop through the loss of poll response from the bypassed station. The loss of polling responses will initiate outage messages which should lead to repair.

The overall goals of these hardware failure considerations is to place the failing station hardware into an off-line non-controlling mode such that no extraneous equipment switching action can possibly occur and to condition the telemetry channel into a bypassed or through-grouped mode so that a degraded local loop telemetry operation can be maintained. Given the total amount of TSC hardware to be distributed over the network, it is also important to carefully consider the overall reliability of the TSC hardware. This is discussed in more detail in Section 11.

9.5.5 Operator Interaction

The entire algorithm is oriented toward supporting the fault isolation task of the tech controller. For any outage,

the algorithm will produce one of two possible results. If automatic restoral is possible, the algorithm will affect this restoral and advise the tech controller of the existance of the failed equipment together with pertinent data regarding the failure mode of the equipment. If the outage is not restorable, the TSC will report a summary alarm together with as much definitive information as possible about the outage.

The TSC supports manual fault isolation by giving the tech controller full access to remote equipment raw data and by providing the controller a remote equipment control capability. Although not detailed in this report, it is also envisioned that the TSC will provide software aids to manual fault isolation in the form of operator "lead-through" diagnostic procedures.

Other specific areas where operator interaction is envisioned are in:

- Manual setting of the failed/not failed status of standby equipment
- Setting of data acquisition masks for alarms to be ignored or not ignored.
- Modifying digroup connectivity tables
- Re-acquiring stream status

10.0 AVAILABILITY IMPROVEMENT

The results of an analysis of the improvement in equipment and circuit availability than can be attributed to TSC control is presented in the section. This analysis was carried out using the failure tree method of calculating equipment unavailabilities (Reference 1), and is based upon the restoral time results presented in Appendix A. These results were derived using a pessimistic model network segment. When applied to the reference circuit used here to compare circuit availability, pessimistic assumptions were also made with respect to the performance of the algorithm, e.g., in every case, it was assumed that the postulated equipment failure occurred at the strategically worst (with respect to algorithm performance) location in the model circuit.

10.1 Equipment Availability

Restoral time data for each equipment type are given in Tabels 10-1 through 10-6. These tables present the outage times for each failure mode and catalog the assumptions that were made regarding restoral actions. Definition of the various failure modes is obtained by referring to the failure trees shown in Figures 10-1 through 10-3. A comparison table is not shown for the TD-1192 since TSC control has no effect on the unavailability of this unit. The unavailability of the TD-1192 both with and without TSC control, as calculated in Reference 1, is $4.36\ (10)^{-5}$.

A comparison of the unavailability of each equipment type with a without TSC control is given in Table 10-7. Note that two figures are given for both the FRC-163 and the TD-1193. These represent optimistic (A) and pessimistic (B) assumptions regarding the amount and reliability of non-redundant circuitry present in each equipment. Note that

the improvement in equipment unavailability depends markedly on the assumptions regarding the probabilities of non-redundant equipment failure. From this, it is concluded that the availability improvement due to control of redundant equipment will be severely limited by the degree of non-redundancy.

In the case of the KG-81/Bypass combination, the improvement limitation is due almost entirely to the two failure modes that involve failure of the bypass. It is believed that the value used for bypass failure probability is perhaps too pessimistic and that the actual improvement in TED availability will be much better. The probability of bypass failure was determined from the formula,

$$S = 1\text{-EXP}\left[-\frac{MTBF_{KG}}{MTBF_{BYPASS}}\right]$$

Being the probability that the bypass will have failed by the time that it is called upon (at the incidence of a KG-81 failure). The values used for KG-81 and bypass MTBF's were 10,000 hours and 120,000 hours respectively. The bypass MTBF value of 120,000 hours is the figure predicted by the equipment contractor for the HNF-81 and includes two bypass switches and the associated AC power supply. Using standard methods, an MTBF prediction was carried out by ECI for a single bypass assuming 48 VDC operation. The resultant MTBF was in excess of 1,000,000 hours. The validity of standard prediction methods for MTBF's this large can be questioned. If is nevertheless felt that a switch failure probability of .08 is too pessimistic and that the actual availability improvement with control will be greater than the values shown in the table.

TABLE 10-1 WALBURN/WALBURN BYPASS OUTAGE ANALYSIS SUMMARY (UNMANNED LOCATIONS)

	Without TSC Control		With TSC Control	
Outage	Value	Condition/Restoral Action	Value	Condition/Restoral Control
ϕ_1	2.6 hrs	Must deduce cause of failure and dispatch a man to the site location to activate bypass	1 sec	Bypass action effected by TSC
ϕ_2	3.5 hrs	Must deduce cause of failure, dispatch man, and repair KG-81	3 hrs	Dispatch man (2.5 hrs) and repair (0.5 hrs)
ϕ_3	2.6 hrs	Same as ϕ_1	2 sec	Isolate and bypass (ϕ_1 + 1 sec isolation time)
ϕ_4	3.5 hrs	Same as ϕ_2	3.5 hrs	Summary alarm, dispatch (2.5 hr) Isolate (30 min) and repair (30 min)

TABLE 10-2 WALBURN/WALBURN BYPASS OUTAGE ANALYSIS SUMMARY (MANNED LOCATIONS)

	Without TSC Control		With TSC Control	
Outage	Value	Condition/Restoral Action	Value	Condition/Restoral Action
ϕ_1	5 min	Alarmed KG-81 failure; bypass activated manually, far end deduces the event and activates far end bypass	1 sec	TSC activates bypass and coordinates far end bypass action
ϕ_2	35 min	Alarmed KG-81 failure, switch fails; repair KG-81 (5 min to determine switch failure)	30 min	TSC alarms switch failure; repair can begin immediately
ϕ_3	10 min	Unalarmed KG-81 failure; operator isolates problem and attempts switching. Switch fails. Operator repairs KG-81 (ϕ_1 plus 5 min isolation time)	2 sec	TSC isolates (1 sec), activates bypass and coordinates far end bypass action
ϕ_4	40 min	Unalarmed KG-81 failure; operator isolates problem and attempts switching. Switch fails. Operator repairs KG-81 (ϕ_2 plus 5 min isolation time)	35 min	TSC unable to restore. Generates summary alarm. Operator isolates (5 min) and repairs (30 min)

NOTES: Probability failure is unalarmed = 0.5 $\quad$ $MTTR_{KG}$ = 30 min
Probability bypass switch fails = 0.08

TABLE 10-3 OUTAGE ANALYSIS SUMMARY
TD-1193 MUX (MANNED)

Without TSC Control			With TSC Control		
Outage	Value	Restoral Action	Outage	Value	Restoral Action
ϕ_1	100 msec	Automatic switchover (spec value)	T_1	0.5 sec	Common equipment failure. TSC algorithm isolates and restores by switching
ϕ_2	6 min	Manual isolation (5 min) plus manual switchover (1 min)	T_2	--	State will not occur. TSC will restore all unalarmed Level 2 mux failures (R=1).
ϕ_3	0	Standby failure (no outage)	T_3	100 msec	Automatic switchover (spec value)
ϕ_9	30 min	Multiple failure - restoral requires repair of one unit	T_4	0	Standby failure (no outage)
			T_7	2 sec	Port failure alarmed by associated Level 1 mux; TSC restores
ϕ_{20}	30 min	Nonredundant failure - restoral requires repair	T_{16}	30 min	Multiple failure - restoral requires repair of one unit (avg MTTR)
			T_{27}	30 min	Nonredundant failure - restoral requires repair (avg MTTR)

NOTES:

C1 = .37450 Common equipment Tx complexity, T1 = .5

C2 = .07775 Common equipment Rx complexity, R1 = .5

$C3 = \begin{cases} 0 \\ .00350 \end{cases}$ Port equipment Tx complexity = .333

P = .03 Port equipment Rx complexity = .667

Q = .03 T = 10,000 hrs

NOTES:

R = 1 (100% TSC restoral of redundant on-line unit failures)

T = 1 (redundant unit switchover once per hr)

D = 0 (no off-line testing by TSC)

TABLE 10-4 OUTAGE ANALYSIS SUMMARY
TD-1193 MUX (UNMANNED)

Without TSC Control			With TSC Control		
Outage	Value	Restoral Action	Outage	Value	Restoral Action
ϕ_1	100 msec	Automatic switchover (spec value)	T_1	0.5 sec	Common equipment failure. TSC isolates and restores by switching
ϕ_2	2.6 hrs	Unalarmed; must isolate (0.1 hr) and dispatch man to switchover (2.5 hrs)	T_2	--	State will not occur. TSC will restore all unalarmed Level 2 mux failures (R=1).
ϕ_3	0	Standby failure (no outage)	T_3	100 msec	Automatic switchover (spec value)
			T_4	0	Standby failure (no outage)
ϕ_9	3 hrs	Multiple failure; must dispatch man (2.5 hrs) and repair one unit (0.5 hr)	T_7	2 sec	Port failure alarmed by associated Level 1 mux; TSC restores
ϕ_{20}	3 hrs	Nonredundant failure; must dispatch man (2.5 hrs) and repair one unit (0.5 hr)	T_{16}	3 hrs	Multiple failure; restoral requires dispatch of man (2.5 hrs) and repair of one unit (0.5 hr)
			T_{27}	3 hrs	Nonredundant failure; restoral requires dispatch of man (2.5 hrs) and repair (0.5 hr).

NOTES:

C1 = .37450	Common equipment Tx complexity, T1 = .5
C2 = .07775	Common equipment Rx complexity, R1 = .5
$C3 = \begin{cases} 0 \\ .00350 \end{cases}$	Port equipment Tx complexity = .333
P = .03	Port equipment Rx complexity = .667
Q = .03	T = 10,000 hrs

NOTES:

R = 1 (100% TSC restoral of redundant on-line unit failures)
T = 1 (redundant unit switchover once per hr)
D = 0 (no off-line testing by TSC)

TABLE 10-5 OUTAGE ANALYSIS SUMMARY FRC-163 RADIO (MANNED)

Without TSC Control			With TSC Control		
Outage	Value	Restoral Action	Outage	Value	Restoral Action
ϕ_1	10 μsec	Automatic switchover (spec value)	T_1	1 min	Common equipment failure. TSC isolates and restores by switching
ϕ_2	6 min	Manual isolation (5 min) plus manual switchover (1 min)	T_2	----	State will not occur. TSC will restore all unalarmed radio failures (R=1)
ϕ_3	0	Standby failure (no outage)	T_3	10 μ sec	Automatic switchover (spec value)
ϕ_9	1 hr	Multiple failure; restoral action is repair of one unit (avg MTTR)	T_4	0	Standby failure (no outage)
			T_7	1 sec	Port failure; TSC restores
ϕ_{20}	1 hr	Nonredundant failure; restoral action is repair (avg MTTR)	T_{16}	1 hr	Multiple failure
			T_{27}	1 hr	Nonredundant failure

NOTES:

C1 = .90 Common equipment Tx complexity, T1 = 0.5

C2 = .03 Common equipment Rx complexity, R1 = 0.5

C3 = $\begin{cases} .01 \\ .001 \end{cases}$ Port equipment Tx complexity, T2 = 0.5

P = .02 Port equipment Rx complexity, R2 = 0.5

Q = .03 T = 10,000

NOTES:

R = 1 (100% TSC restoral of redundant on-line unit failures)

T = 1 (redundant unit switchover once per hr)

D = 0 (no on-line testing by TSC)

TABLE 10-6 OUTAGE ANALYSIS SUMMARY FRC-163 RADIO (UNMANNED)

Without TSC Control			With TSC Control		
Outage	Value	Restoral Action	Outage	Value	Restoral Action
ϕ_1	10 μsec	Automatic switchover (spec value)	T_1	1 min	Common equipment failure. TSC isolates and restores by switching
ϕ_2	3 hrs	Isolate remotely, dispatch man and switchover	T_2	---	State will not occur. TSC will restore all unalarmed radio failures (R = 1)
ϕ_3	0	Standby failure (no outage)	T_3	10 μsec	Automatic switchover (spec value)
ϕ_9	4 hrs	Multiple failure; isolate (remotely), dispatch man and repair	T_4	0	Standby failure (no outage)
			T_7	1 sec	Port failure, TSC restores
ϕ_{20}	4 hrs	Nonredundant failure; isolate (remotely), dispatch man and repair	T_{16}	3.5 hr	Multiple failure. Isolate with TSC aid, dispatch man and repair
			T_{27}	3.5 hr	Nonredundant failure. Isolate with TSC aid, dispatch man and repair

NOTES:

C1 = .90	Common equipment Tx complexity, T1 = 0.5
C2 = .03	Common equipment Rx complexity, R1 = 0.5
C3 = $\begin{cases} .01 \\ .001 \end{cases}$	Port equipment Tx complexity, T2 = 0.5
P = .02	Port equipment Rx complexity, R2 = 0.5
Q = .03	T = 10,000

NOTES;

R = 1	(100% TSC restoral of redundant on-line unit failures)
T = 1	(redundant unit switchover once per hr)
D = 0	(no off-line testing by TSC)

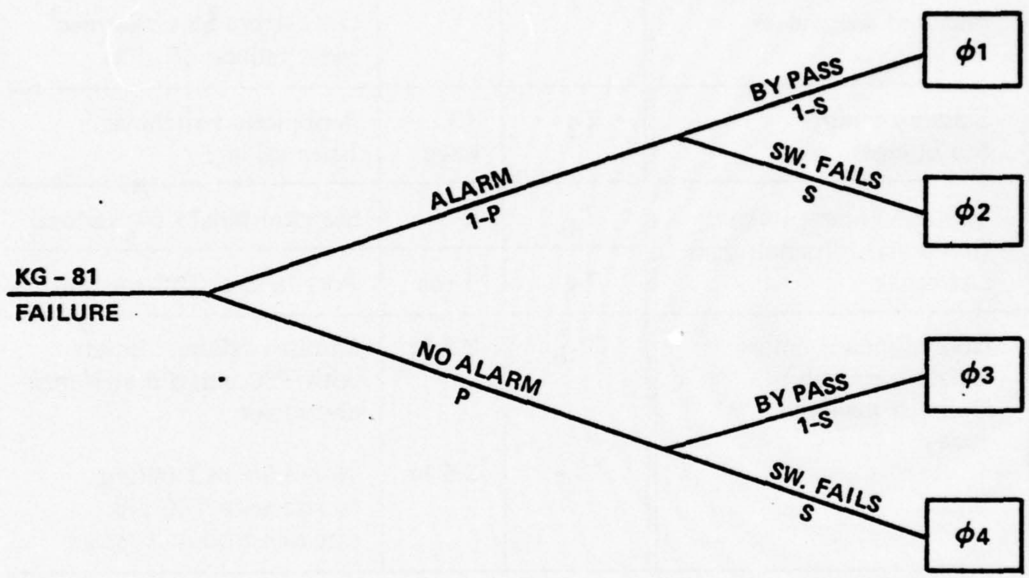


FIGURE 10-1

WALBURN/WALBURN BYPASS FAILURE TREE

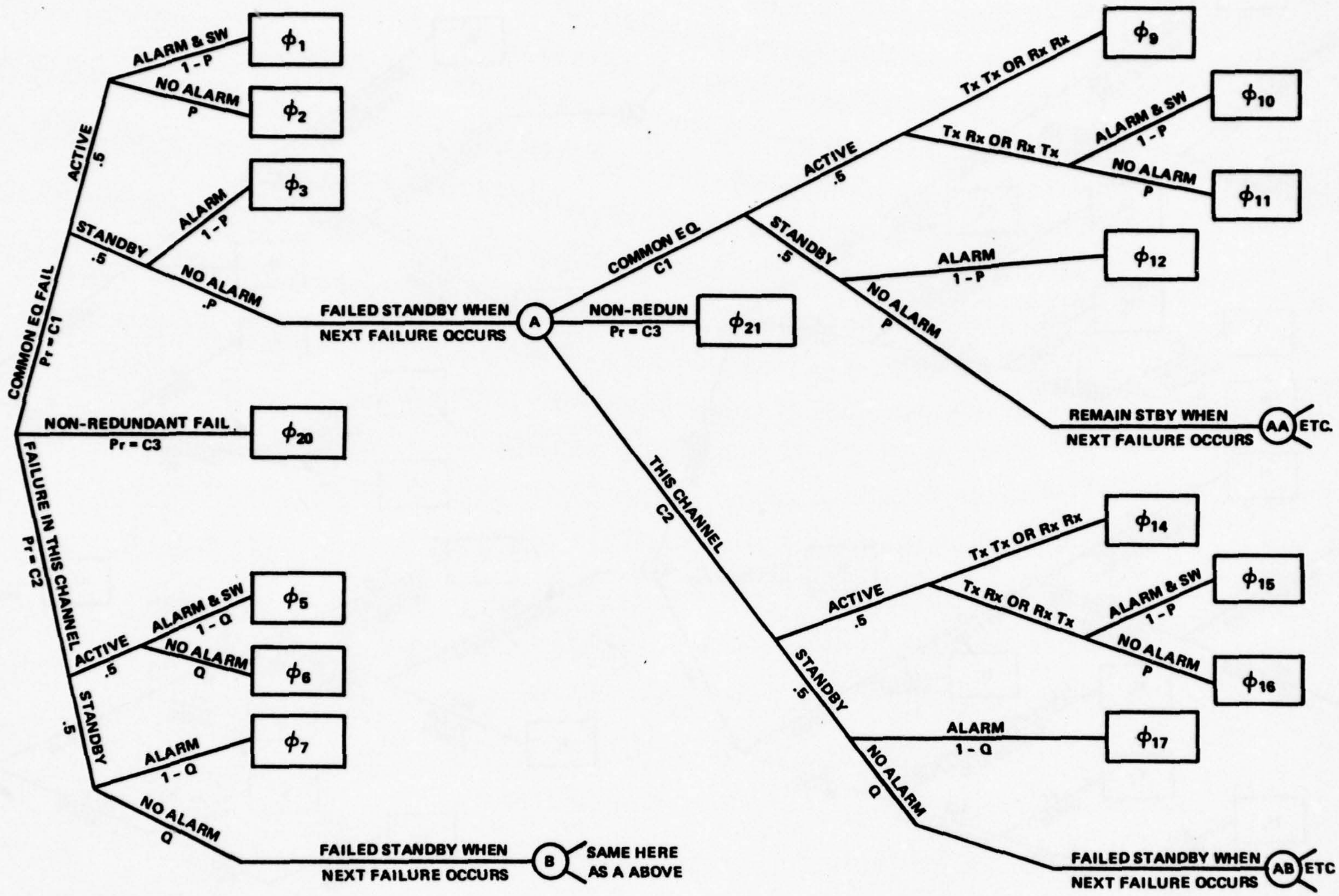


FIGURE 10-2

RADIO AND LEVEL 2 MUX FAILURE TREE - NO TSC

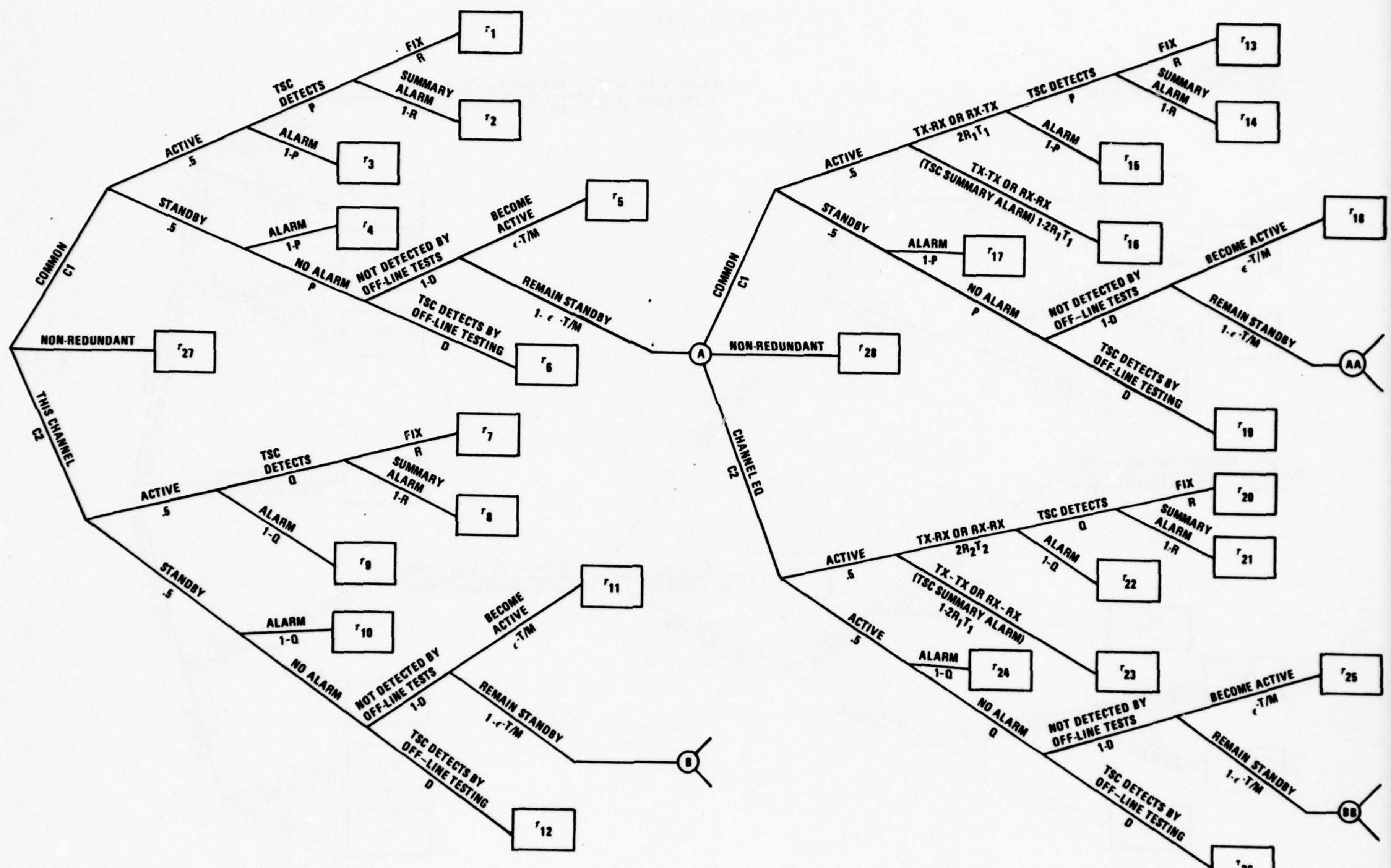


FIGURE 10-3

RADIO AND LEVEL 2 MUX FAILURE TREE - WITH TSC CONTROL

TABLE 10-7 EFFECT OF TSC CONTROL ON EQUIPMENT AVAILABILITIES

Equipment	No TSC	With TSC
TD-1192	4.36 $(10)^{-5}$	4.36 $(10)^{-5}$
TD-1193 (Manned)	(A) 6.83 $(10)^{-7}$	(A) 5.82 $(10)^{-9}$
	(B) 1.77 $(10)^{-6}$	(B) 1.10 $(10)^{-6}$
TD-1193 (Unmanned)	(A) 1.27 $(10)^{-5}$	(A) 6.59 $(10)^{-9}$
	(B) 1.91 $(10)^{-5}$	(B) 6.57 $(10)^{-6}$
KG-81/Bypass (Manned)	1.65 $(10)^{-5}$	4.37 $(10)^{-6}$
KG-81/Bypass (Unmanned)	2.67 $(10)^{-4}$	2.60 $(10)^{-5}$
FRC-163 (Manned)	(A) 2.78 $(10)^{-6}$	(A) 7.15 $(10)^{-7}$
	(B) 8.23 $(10)^{-6}$	(B) 6.34 $(10)^{-6}$
FRC-163 (Unmanned)	(A) 2.82 $(10)^{-5}$	(A) 2.28 $(10)^{-6}$
	(B) 4.84 $(10)^{-5}$	(B) 2.20 $(10)^{-5}$

NOTES:

(A) Probabilities of Nonredundant Circuitry Failure: TD-1193 = 0; FRC-163 = 0.001

(B) Probabilities of Nonredundant Circuitry Failure: TD-1193 = 0.0035; FRC-163 = 0.01

10.2 Circuit Availability

In order to determine the effect of circuit availability, a reference circuit was chosen. The circuit selected for this comparison is one that extends from Shape to Hillingdon. This circuit was picked because it contains every equipment configuration of interest and has a chain of five unmanned repeaters. Since the TSC availability improvement is greater for unmanned equipment configurations, the results obtained using this reference circuit will reflect the effects of control in its best light. For the more common DEB circuit configurations, the improvement will not be as great.

Results were calculated for both degress of non-redundant circuitry assumed for the TD-1193 and the FRC-163. These results are shown in Tables 10-8 and 10-9. For the case of the more optimistic non-redundancy assumptions, the circuit availability improvement is nearly an order of magnitude.

Examination of the contributions due to each equipment type shows that without TSC control, the largest single contributor is the unmanned Walburn. TSC control reduces this contribution by more than an order of magnitude (even assuming a .08 probability of bypass failure). The contributions of the next largest contributor (the unmanned radio) is also reduced by more than an order of magnitude. Note that for the more pessimistic assumptions regarding the degree of nonredundant equipment (Table 10-9), the reduction in the unavailability contribution of the unmanned radio due to control is only slightly more than a factor of 2.

TABLE 10-8 CIRCUIT UNAVAILABILITY SUMMARY
C3 (TD-1193) = 0.0035
C3 (FRC-163) = 0.01

Equipment	CIRCUIT UNAVAILABILITY CONTRIBUTION No TSC	With TSC
(2) TD-1192	$8.72 (10)^{-5}$	$8.72 (10)^{-5}$
(2) TD-1193 (M)	$3.54 (10)^{-6}$	$2.20 (10)^{-6}$
(2) TD-1193 (U)	$3.82 (10)^{-5}$	$1.31 (10)^{-5}$
(2) KG-81 (M)	$3.30 (10)^{-5}$	$8.74 (10)^{-6}$
(2) KG-81 (U)	$5.34 (10)^{-4}$	$5.20 (10)^{-5}$
(4) FRC-163 (M)	$3.29 (10)^{-5}$	$2.54 (10)^{-5}$
(12) FRC-163 (U)	$5.81 (10)^{-4}$	$2.64 (10)^{-4}$
Total Circuit Unavailability	$1.31 (10)^{-3}$	$4.53 (10)^{-4}$

TABLE 10-9 CIRCUIT UNAVAILABILITY SUMMARY
C3 (TD-1193) = 0
C3 (FRC-163) = 0.001

Equipment	CIRCUIT UNAVAILABILITY CONTRIBUTION No TSC	With TSC
(2) TD-1192	$8.72 (10)^{-5}$	$8.72 (10)^{-5}$
(2) TD-1193 (M)	$1.37 (10)^{-6}$	$1.16 (10)^{-8}$
(2) TD-1193 (U)	$2.54 (10)^{-5}$	$1.32 (10)^{-8}$
(2) KG-81 (M)	$3.30 (10)^{-5}$	$8.74 (10)^{-6}$
(2) KG-81 (U)	$5.34 (10)^{-4}$	$5.20 (10)^{-5}$
(4) FRC-163 (M)	$1.11 (10)^{-5}$	$2.86 (10)^{-6}$
(12) FRC-163 (U)	$3.38 (10)^{-4}$	$2.74 (10)^{-5}$
Total Circuit Unavailability	$1.03 (10)^{-3}$	$1.78 (10)^{-4}$

Considering that the results shown for the Walburn are believed to be markedly pessimistic, it is fair to conclude that the improvement in availability afforded by the TSC is limited by 1) the presence of the non-redundant TD-1192, and 2) the degree of non-redundant circuitry in the TD-1193 and the FRC-163.

Although remote control of the Walburn bypass and redundant equipments does not provide spectacular gains in circuit availability, these capabilities are almost essential from an operational standpoint. Without visibility of remote alarms and remote control, long circuit outages can be experienced and a great deal of manpower wasted in trying to determine the location of a failure and traveling to the remote site to perform such a simple task as bypass activation.

Remote manual control as opposed to automatic would solve the basic problem of bypass activation with very little added penalty in terms of unavailability. Automatic fault isolation, however, is also of great operational value. Most important is the decrease in manpower and skill level that it affords by pinpointing faults and the elimination of contention and futile efforts that it provides through outage notification to alarming stations that really have no problem.

11.0 IMPLEMENTATION

11.1 Hardware Design

Numerous hardware architectures have been considered for implementation throughout the course of this study. All have involved a partitioning of the functions required for a working TSC system between hardware and software (or, more generally, processor control) and for those functions which are amenable to processor control, a partitioning between processors.

Each major function must be considered separately then each block and function must be considered in relationship to other functions and to the system as a while. Also, there is a need to consider both stressed and unstressed operation and how the overall cost and performance of the system is affected as the performance of each function is improved or degraded. From all of these characteristics, a set of compromises is requried to arrive at a final hardware architecture.

11.1.1 General Design Considerations

Some general rules of thumb that are helpful in evaluating some of the trades to be made are as follows. First, hardware (combinational logic) is almost invariably faster than software. Second, software is usually less costly in complex decision processes where the time required is tolerable. Third, a processor can serve more than one function within a system, but not simultaneously. Fourth, in a request-response processing mode, the average duty cycle (or utilization) of a processor should be maintained at about 50% and no greater than 70% if extensive queueing and delays are to be avoided.

Definitions of the normal and stressed operation of the system are necessary to evaluate the results of the subsequent analysis. The normal operating mode is characterized by all

equipment within a local loop in a known operational status, either failed or not failed. Within the domain of a local loop, which includes both the loop equipment and the equipment affecting the streams of a loop, failures are infrequent, compared to the loop response time. This yields a normal operating mode of no change in the status of equipment alarms and simple polling on the telemetry channel.

The most common stressed mode is the failure of a single piece of equipment within the domain of a local loop. This results in an automatic fault isolation and service restoral attempt and potential operator involvement until service is restore.

The occurrence of stressed modes represent occasional intrusions on the normal operating mode. This suggests that a system optimized for a stressed mode of operation will be grossly under utilized in the normal operating mode. A system optimized for the normal operating mode will present a bottleneck during stress periods.

Each of the four major areas of the system (Telemetry, Data Acquisition, System Functions, and CDU) are subject to hardware/software trade-offs. An initial review of the overall functions within the area suggest that the sytem functions will likely be software due to the large diversity of the functions and the complexity of the decision processes. At least certain portions of the telemetry and data acquisition functions are potential candidates for hardware implementation due to speed considerations, the availability of LSI parts for the functions, or simplicity of the operations to be performed. The CDU functions has certain sub-functions that are potential software candidates due to the complex interaction between operator and system.

One major goal for the telemetry system is to minimize access delay for both the stressed and normal operating modes. The access delay of the telemetry channel and telemetry function is one major determinant of the polling rate during the normal operating mode. During stressed periods, the telemetry channel access delay again plays a strong role in determining system response time.

Another characteristics of the telemetry channel is the anticipated telemetry channel traffic. A large portion of the stressed mode traffic is typically generated in bursts. These bursts will have messages of variable length. Interloop traffic will, in general, require buffering. The inclusion of this buffering as part of the telemetry channel hardware as opposed to main memory is a definite alternative.

Data acquisition of critical alarm data is best performed on a report by exception basis. In order to determine this change in data acquisition status so that it might be reported, the raw information must be continuously scanned by either hardware or software and compared to previously stored values. The response rate to changes in equipment state is also of some concern since it is this response latency that determines how quickly such changes are available to the TSC system.

It should also be noted that the expected status changes will also occur in bursts. A large number of the anticipated failures within the system have a variety of alarms that are associated with them. A radio fade on a branch which is fully populated at one site can easily generate more than 50 primary alarms. Again a buffering function such that no alarm changes are lost is required.

Also required as part of the data acquisition function is the availability of raw data. It is envisioned that raw data requests will usually be manually generated. However, at least certain portions of the raw data are required by the automatic fault isolation and restoral algorithm. The use within the automatic fault isolation and restoral algorithm is to determine the state of the equipment before and after switching actions.

Control display functions require careful consideration. Large panels of indicators and switches are costly and not easily changed to accomodate connectivity changes. A potentially more useful arrangement is to provide more concise continuously updated change information to the personnel and provide any additional information that may be required through simple means at their request.

11.1.2 System Architecture

Having derived some gross functional requirements for the various TSC functions, numerous hardware/software architectures were explored on a fairly high level. The range of these architectures was from complete hardware to systems employing multiple processors. Two general observations from these gross evaluations are of importance: a totally hardware system will not provide the same degree of flexibility as a processor-based system at comparable costs; the use of multiple processors substantially increases the hardware complextiy over a single processor.

Before exploring the detailed architectural trades, some additional observations are useful. An ideal system would allow restructuring of its capabilities to meet the instantaneous needs of the network. During periods of the normal operation, the system would be optimized for routine tasks. During stress periods, restructuring to support the stressed situation would occur.

During periods of normal mode operation, no problems exist in a dynamic system. Only one process is active and the whole of the resources can be involved in this task. If one or more branches is stressed, then two processes become active. There is an obligation to continue the normal mode on all branches and a need to allocate resources to the stress situation. This suggests that a dynamic allocation will degrade the normal operating mode during stress. This is not an intolerable situation if the normal mode can be maintained above a minimum required value and the degree of degradation can be bounded and controlled.

One additional highly desirable feature is common system hardware and software. An ideal system would employ common hardware and software across the range of sites within the system. While this may create some underutilization at simple stations, the overall life cycle cost is reduced. Non-recurring costs are minimized, spare parts inventories are minimized and personnel training is reduced.

Examining each of the four functional requirements with reference to the system requirements, additional constraints can be derived that further restrict the architecture. Some of the following observations are not only conclusions which can be derived but it is believed that they are valid for the situation.

Data acquisition requires a scanning function as opposed to an asynchronous change detection system. Further, if this scanning function is performed by a microprocessor, the entire capabilities of a single microprocessor are required to give a reasonable level of performance. The overall scanning operation is very simple and lends itself well to a hardware implementation. The more complex processes involved in data acquisition are only a small part of the overall function and can be performed as part of the system functions. The availability of LSI first in-first out memories allow the inclusion of hardware buffering with the hardware scanning function.

The availability of recently introduced multi-protocol LSI USRTs definitely suggest that the majority of the line protocol function is well handled in hardware. Further, the availability of very low cost random access memories allow the simple, low cost implementation of first in-first out memories which would allow a hardware buffer as part of the telemetry channel hardware. Given the line data rate and assuming that the hardware stores the telemetry channel information in a byte parallel form, the requirements for the memory are very non-critical.

CDU functions, especially requests for information and control by operations personnel, are likely best served by a terminal or terminal like device. The processing load

involved in the display of information in an easily interpretable form can be significant. Availability of commercial intelligent terminals suggest that the majority of these display formatting functions be performed within the terminal.

As previously indicated, the majority of the system functions are candidates for processor implementation because of the complex decision processes involved. Major simplification of the system functions can be gained through the selection of a processor which is well suited to the anticipated environment and organization of the other system hardware.

From all of the preceding discussion along with information throughout this report, the architecture of Figure 11-1 has been derived. In general, both the overall architecture and the detailed implementation is conceptually simple. This should yield a low cost system which remains a primary consideration.

The autonomy of the modules and use of a single processor allow the potential for a dynamic allocation of processing resources. In fact, this dynamic allocation is mandatory with a single processor. Even though the appearance of multiple function can be generated, a single processor can only be involved in a single function during any instant in time.

Given the normal operating mode and dynamic allocation of processing resources, an interrupt driven hardware and software architecture is an obvious choice. However, there is both a hardware and software overhead associated with this type of implementation. In order to appropriately specify such an architecture either the benefits to be gained from such an architecture must be much greater than reasonable alternatives or the hardware and software overhead must be less than the alternatives.

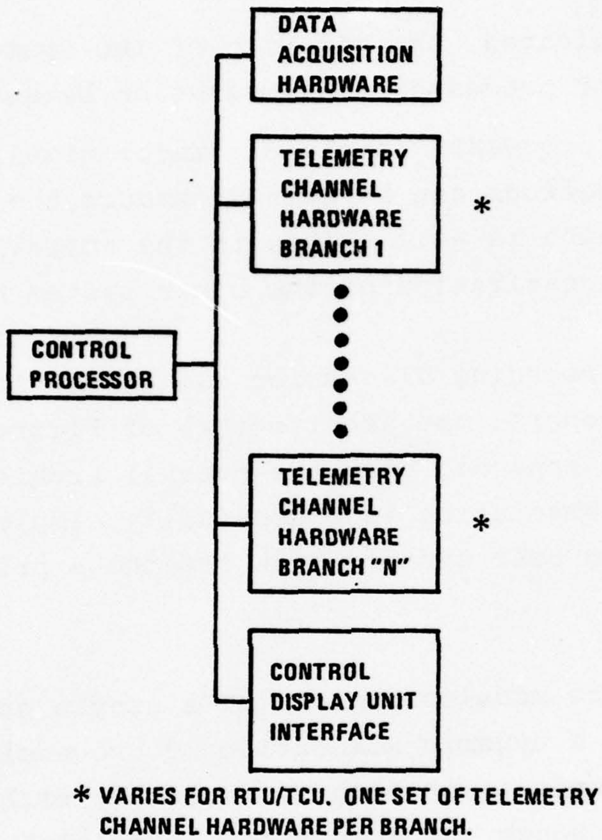


FIGURE 11-1

TCU/RTU ARCHITECTURE

The primary alternatives to an interrupt driven single processor are a single processor polling system or a multiprocessor polling or interrupt driven system. The polling system, either single or multiprocessor, requires less hardware. Each function that potentially requires service is periodically sampled and a determination is made as to the need for service. If service is needed, it is initiated and as soon as service is completed, polling is resumed. There are at least two major difficulties in a polling system. First, substantial amounts of processing resources can be consumed in the polling function. Second, once service is initiated, polling ceases and potentially high priority requirements are delayed.

Virtually any multiprocessor system is more complex than a single processor system. A software base is required which forms the actual application program. For practical considerations, this is fixed independently from the number of processors employed in a system. In a multiple processor system, additional software is required to coordinate the flow of information between the processors. Also, there is an additional amount of hardware for this exchange of information.

Hardware requirements for an interrupt driven system include the additional hardware to generate and capture interrupts and any memory required for interrupt service overhead. Signals usually exist or can be simple generated for interrupts. Some additional logic is usually required to meet the interrupt structure needs of the processor. This is typically six or less SSI parts. For the processor to be discussed, the priority interrupt structure can be implemented with 4 parts: two SSI parts; and two MSI parts. Thus the overall hardware needed to implement an interrupt system is small.

Software requirements for an interrupt driven system consist of the interrupt prologue and the interrupt return epilogue. The prologue is concerned with saving the current running state of the processor so that when interrupt service is complete, the processor can return to the function it was performing prior to the interrupt. The epilogue restores the previous program.

There are two vital concerns for the prologue and epilogue. First, the amount of memory required for both, which includes the software instructions or program required and the amount of storage required to execute the prologue and epilogue. If the time required is excessive then a polling discipline may be appropriate. Further, any time involved in the prologue or epilogue is overhead and detracts from the overal system performace. In general, the overall memory requirements associated with the prologue and epilogue is reasonably small compared to the magnitude of the application software independently of the processor examined. The time involved is not and varies significantly between processors.

Several commercially available SCADA systems were evaluated with respect to their appropriateness within the DEB network. For a variety of reasons none of the system evaluated are directly applicable. Most commercially available systems provide very sophisticated analog processing and control capability with only minor consideration to digital I/O. Further, a majority of the systems are designed for fixed, in-plant installation and require multiple conductor cabling.

Questions directed to the manufacturers of these equipments confirmed its inability to be modified for use with the telemetry channel.

Those manufacturers who produce remote systems which could potentially be used within the context of the DEB network were primarily concerned with operation over leased lines at data rates of up to 2400 baud. Again, it was confirmed that these rates could not be increased to be compatible with the telemetry channel. The limitation of 2400 baud of these systems would limit their performance substantially and are thus not suitable for performance comparisons.

Current state-of-the-art processing techniques include a great deal of activity in the area of multiprocessor, distributed architectures. This was considered in some detail prior to arriving at the suggested implementaion. Ignoring the cost of such a system, a reasonable distributed architecture may incorporate a single processor for each of the four functional areas. If some optimistic assumptions are made, a performance for a state of the art system can be derived. Assume that the normal operating mode represents a base level for the distributed system. This assumption is not invalid because of the distribution of processing to support the normal mode operations. If we assume the processing associated with stressed mode is distributed equally between the four functions (or processors) then the total effective processing time can be reduced by a factor of four.

Elsewhere in this report (Appendix A) restoral times for various failure modes are derived. Shown with these restoral times are peak and average processor utilizations over what has been referred to here as a stressed operation mode. The peak utilizations ranged from a high of 209 ms to a low of 45 ms for these failures. Given the distributed architecture, the peak utilizations would range from 52 ms to 11 ms.

The overall restoral times would not be grossly affected. The major reason for this is that at the processing rates being dealt with, the major driving force is the equipment resynchronization time. Realistic assumptions concerning the operation of a distributed architecture system would probably yield restoral improvements of less than 20%. Furthermore, a majority of the processing capability with this distributed system is grossly underutilized.

11.1.3 Processor Implementation

Processor selection for this system can be summarized as follows. A majority of what can be generically defined as minicomputers can be eliminated from consideration in a similar way in which a distributed architecture can. In general, the processing capability greatly exceeds the need. The next set of candidates to be considered are top end microprocessors. Within this domain, some of the potential candidates are the Intel 8080, Motorola 6800, and Texas Instruments 9900.

An analysis of the overall system software based upon typical programming techniques was conducted at a high level to attempt to derive the high frequency operations which would affect software execution time. Excluding the CDU from the entire analysis, the following general characteristics were derived. First,for the interrupt driven system,the task organization of the software and the anticipated large use of subroutines suggest that context swtiching will be very frequent. Second, array addressing will also be very common. Third, pointers will be frequently used and another level of indexing is important. Fourth, there will be substantial amounts of data movement to and from memory and character manipulation within memory.

Comparisons of microprocessor performance against these software functions was conducted. Both internally created benchmarks and supplied benchmarks were used. From these software considerations, the Texas Instrument's TMS 9900 emerged as the mest choice for the system. Further investigation of this microprocessor yielded additional positive attributes. First, it is part of an integrated family which contains significant software support. Second, speed enhanced I^2L mircoprocessors are planned for introduction well within any period of anticipated deployment of the TSC system. These processors will offer a 2.5 increase in processing rate. Third, the 16 general registers within the processor are sufficient to contain all of the required variables for many functions. This will reduce program size and decrease processing time simultaneously. Interrupt response and context switching performance is illustrated in Figure 11-2.

The proposed processor architecture is shown in Figure 11-3. In general, it is very straight-forward in terms of the connection of the blocks for the processor. Three major blocks require some additional explanation and rationalization. Forward error correcting coding is shown as part of the memory module. The use of the watchdog timer and non-volatile storage must also be explained.

A requirement for reliable and predictable operation for the TSC hardware has been defined in several locations within this report. The arguments for this reliability can be summarized. First, reliability is required for the purposes of control functions. In appropriate control caused by TSC hardware failures must be prevented. Second, the costs related to maintenance are reduced by reliable equipment. Third, the repair and maintenance of other network equipment should not be degraded by unavailability of the TSC hardware.

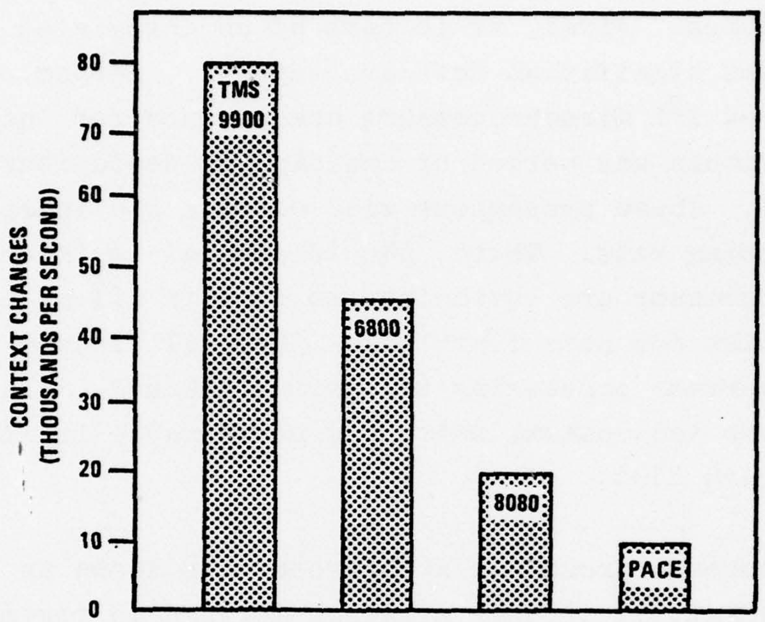


FIGURE 11-2

COMPARISON OF MICROPROCESSORS - CONTEXT SWITCHING

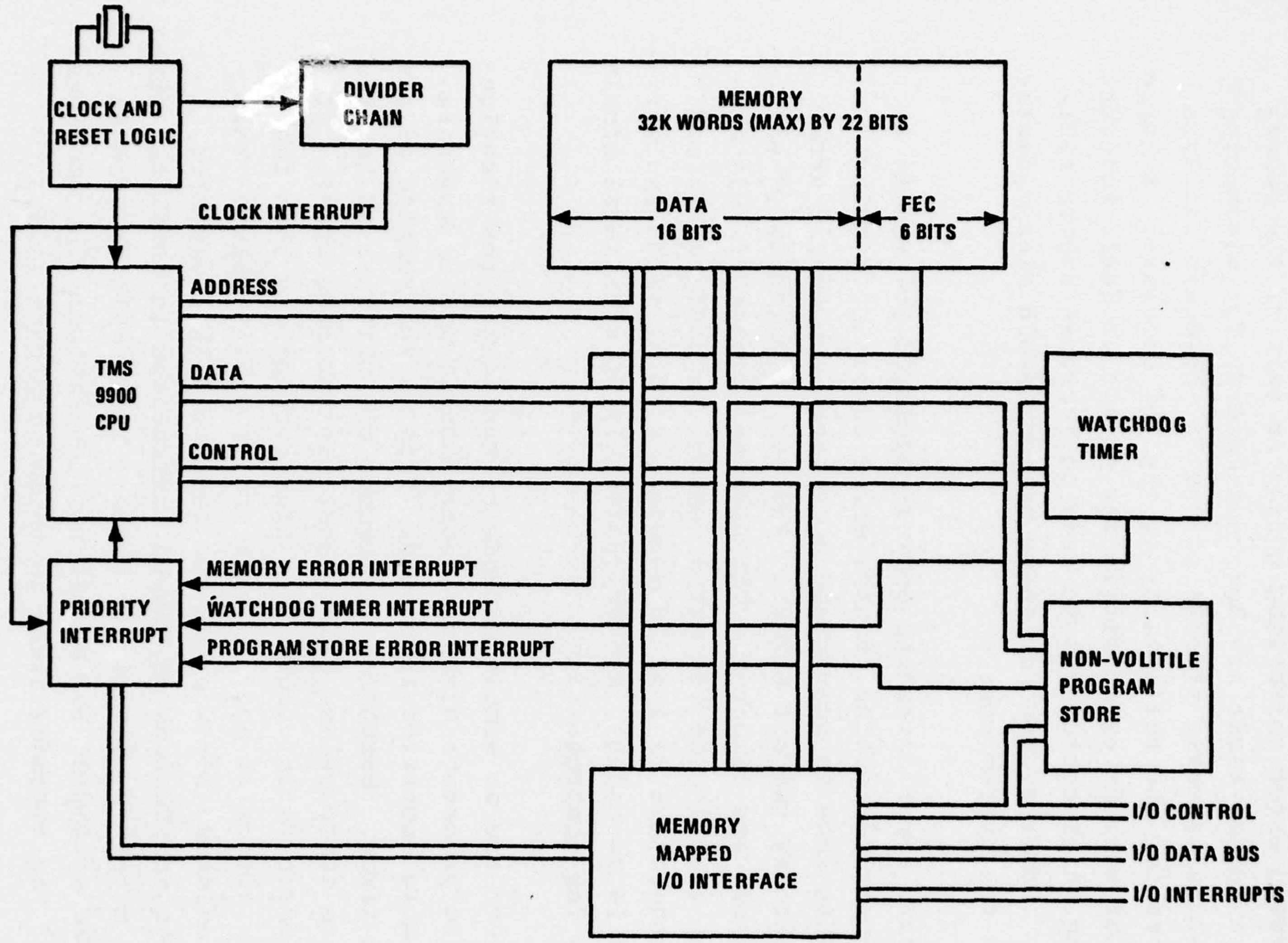


FIGURE 11-3
PROCESSOR BLOCK DIAGRAM

Both forward error correcting coding as part of the memory and the watchdog timer are low cost methods for minimizing any potential adverse effects of the TSC hardware failures and increasing the reliability of the TSC hardware. A large variety of methods exist within the domain of fault tolerant computing which could also be used to increase system reliability. However, most of these methods would significantly increase the system cost.

The forward error correctin coding suggested for use in a single error correcting, double error detecting Hamming code. This code is guaranteed to correct any single error and detect any two bit errors. Further, a majority of all other errors are detected. The hardware impact of this code, above the addition of the extra memory, is confined to six parity generators, a 1 of 16 decoder, a total of 16 exclusive-or gates (4 packages), and approximately 10 additional simple packages for control.

Since there are no simple methods of monitoring the functioning of the processor directly, some other means of insuring its reliable operation is needed. This is the purpose of the watchdog timer. Scattered throughout the software at critical points are instructions which reset the watchdog timer. If the processor fails, there is a high probability that the watchdog timer will not be reset and it will timeout. This becomes evidence for a processor failure. The suggested implementation decodes an unused instruction to reset the timer. A set of metallic contacts is shown. These will be used to bypass TSC equipment for a fail soft operation. The implementation of the watchdog timer is shown in Figure 11.4.

Several alternatives exist for overall memory implementation. First, a predominantly Read Only Memory (ROM) implementation with Random Access Read/Write Memory (RAM) used only for non-critical or volatile variables is possible. A second alternative is to store a majority of common software in ROM with

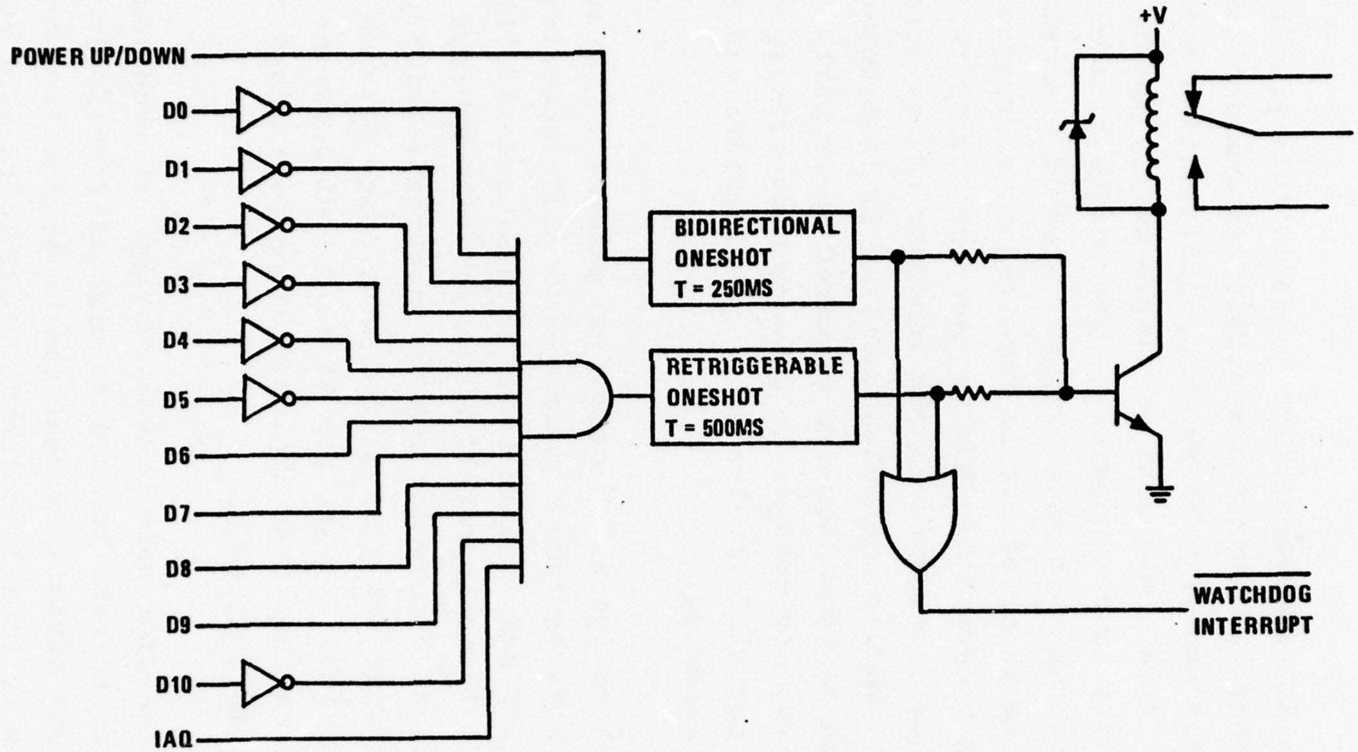


FIGURE 11-4
WATCHDOG TIMER

station specific software in Ram. The third possibility is a totally RAM system.

A predominantly ROM implementation is definitely possible. All common software and station specific information can be contained in this manner. Three major difficulties exist with this approach. First, any goal of simple memory module interchangeability is not possible. The unique station information eliminates this possibility. Even if programmable ROM is used, some major effort is required to reprogram. Second, if mask programmable ROM is used for the software base (as would be considered in the interest of low cost implementation), module interchangeability is lost between RTUs and TCUs which increases the required spare parts inventory. Third, any changes in network connectivity or control philosophy will involve substantial effort and costs within the TSC hardware.

Placing any information which cannot be generated from available information such as equipment complements and connectivity into semiconductor RAM requires some form of non-volatile backup storage. If an appropriate non-volatile storage is chosen, a potential exists for module interchangeability. Some advantages that are gained from this usage outside of module interchangeability include simpler modifications to station data bases and lower implementation cost since the variety of ROMs is reduced.

Given a requirement for some backup storage, it requires only simple extensions to make the system virtually entirely RAM. There are numerous advantages that can be gained from this. First the system is totally software flexible to meet changing requirements. Second, total addressable memory can be reduced since overlay techniques can be used. For example, the total

power up initialization task is only used on start-up. After it has performed its task, the memory occupied by this task can be used for other system tasks. Similar considerations exist for other infrequently used functions such as diagnostic software and certain maintenance aid functions. Third, common memory hardware modules can be used in all stations. The basic memory module is illustrated in Figure 11-5.

The non-volatile storage requirement can be accomplished in a variety of ways. There are a number of different storage media and a number of different ways of deploying this storage. From strictly reliability considerations, the use of any mechanical media becomes a second choice as opposed to any non-mechanical media. From survivability considerations, distributed storage is preferable to any centralized storage. One situation seriously considered confined the back-up storage to a series of centralized locations which could down-load software to the stations via the telemetry channel.

The suggested implementation distributes the non-volatile storage to each processor in the form of a small magentic bubble storage array. The details of this implementation are shown in Figure 11-6, which has been taken directly from Texas Instruments application notes. Given requirements for tagging information to identify information within this storage, the useful capacity of this memory is on the order of 10,000 bytes. This amount of storage may not be adequate to contain all of the operational software and diagnostic support software. A second bubble memory package can be added which will double the capacity to 20,000 bytes. This should be more than adequate. (This may sound expensive, but TI is presently offering 20,000 byte bubble memory modules for their Model 765 terminal at $500 each.)

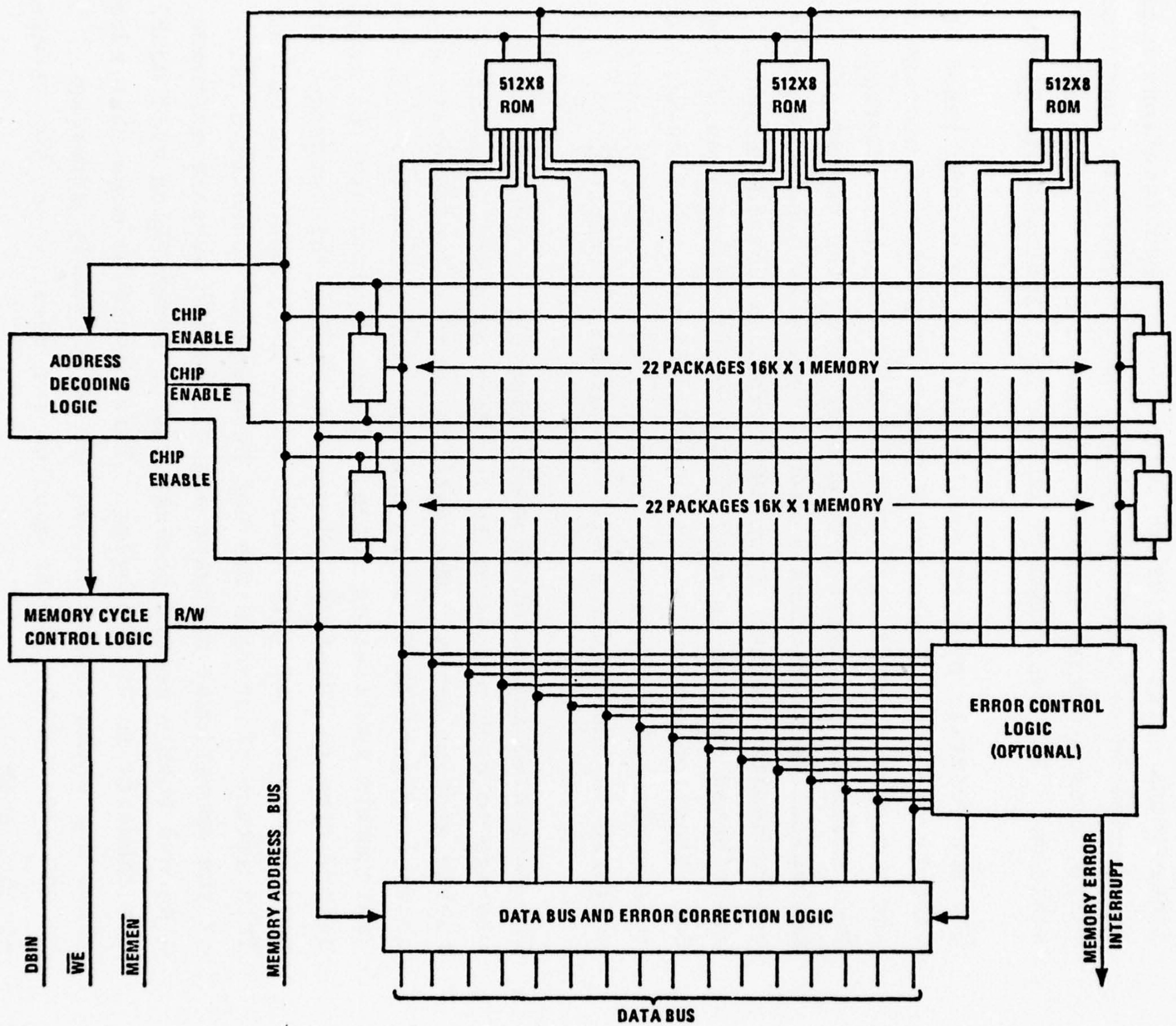


FIGURE 11-5
RANDOM ACCESS MEMORY MODULE

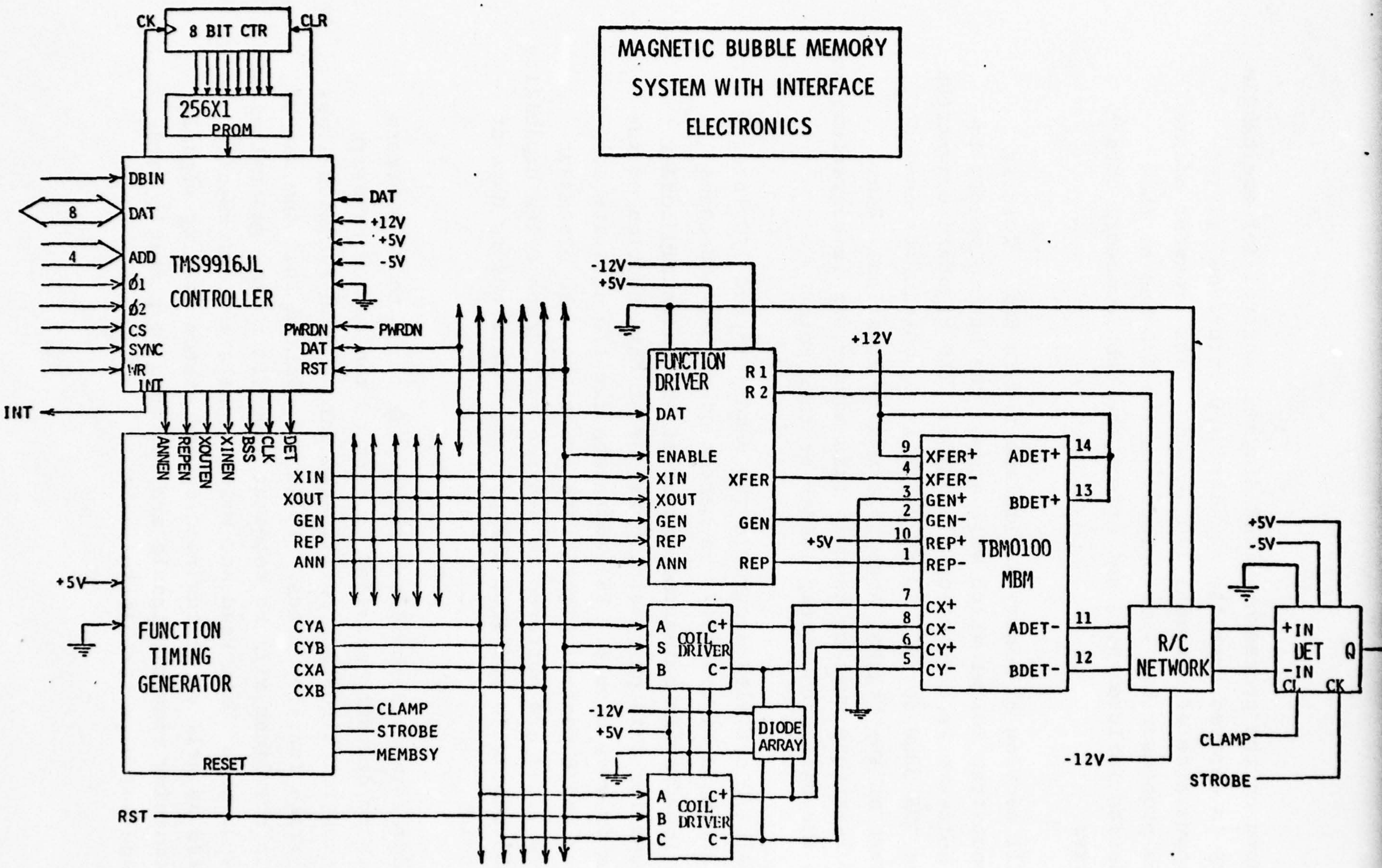


FIGURE 11-6
MAGNETIC BUBBLE MEMORY

As shown on the processor block diagram, control for the bubble memory is derived from the standard I/O interface, giving the appearance of a normal I/O device. Data transfer occurs on the processor bus - as opposed to the I/O bus so that I/O device failures will not affect the bubble memory data transfer.

A small section of memory is implemented in ROM. This is the bootstrap loader which will access the bubble memory to load software in a power-up situation. The standard convention of the TMS 9900 for a load sequence is to obtain the needed pointers at the highest available memory location. Since these locations must be used in this manner, it is convenient to place the entire bootstrap loader at this point.

General I/O activity occurs in the memory mapped I/O interface shown in Figure 11-7. A series of memory addresses taken immediately below the extent of the bootstrap loader are used for this purpose. Operation of this section of the system is very simple. The addresses for I/O activity are decoded and 8 control signals are generated, the processor address lines are buffered to provide sufficient drive capability and the gating to and from the processor data bus is done at this point.

The addressing scheme allows addressing of up to 32 discrete devices by decoding provided as part of the device hardware. The 8 control lines are partitioned into 4 read function lines and 4 write function lines. It is anticipated that the use of the control lines will be somewhat specific to the general class of the device. Data read and write signals are universally required as well as status reading and status setting signals. The remaining control signals are required to simplify hardware and software interactions.

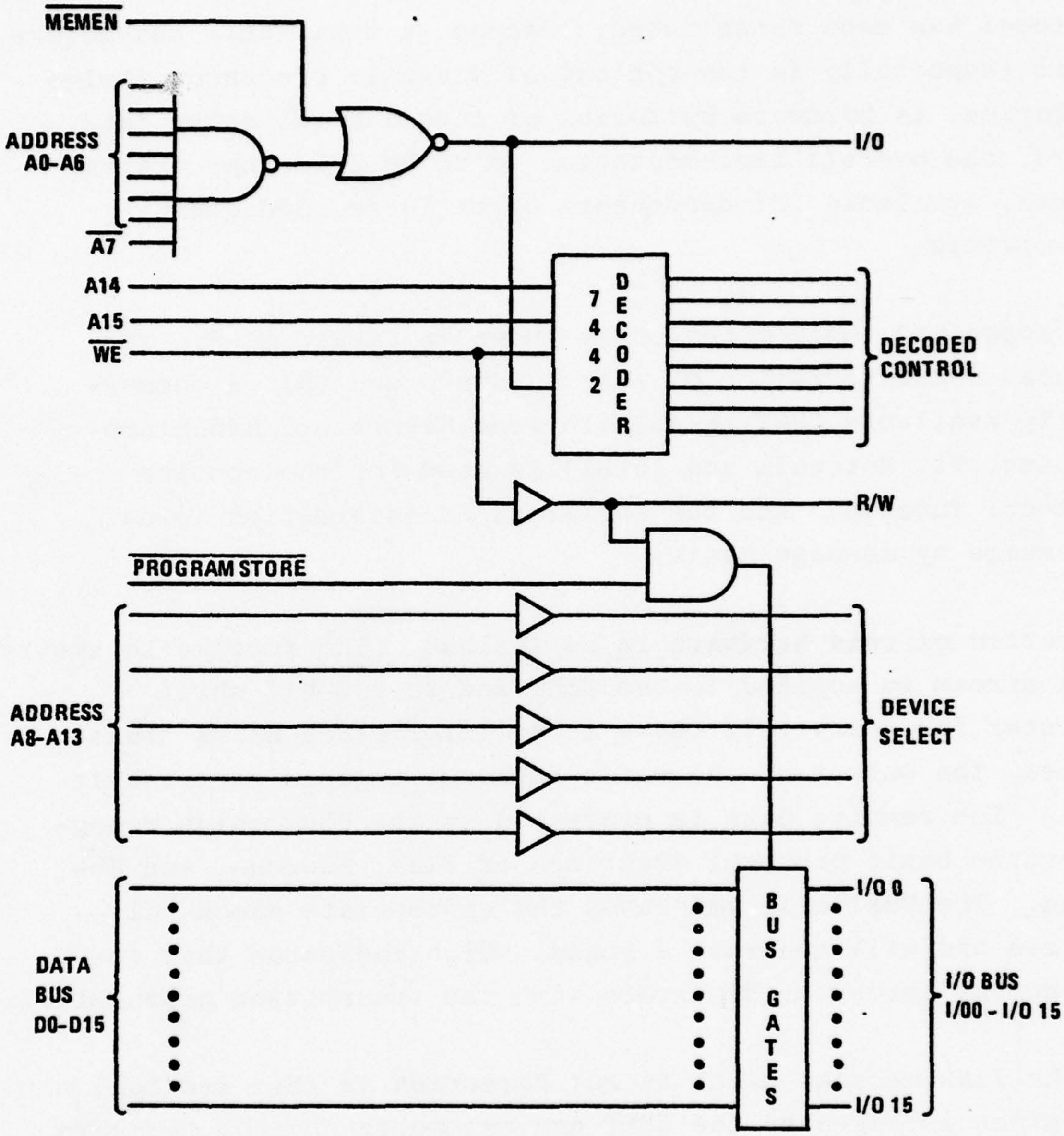


FIGURE 11-7
MEMORY MAPPED I/O

11.1.4 Telemetry Subsystem

Some of the requirements for the telemetry channel hardware have already been at least partially defined. First link protocol has been established. Second, a desireable characteristic (especially in the context of a single processor implementation) is hardware buffering of incoming telemetry data. Third, the overall implementation is to be interrupt driven. Fourth, available LSI components ought to be used where appropriate.

The suggested implementation is shown in Figure 11-8. The general characteristics of this hardware are that a commercially available USRT (available from Signetics, SMC microsystems, TI, Motorola and Intel) is used for the routine protocol functions and the buffering of information is on a message by message basis.

Operation of this hardware is as follows. The receive telemetry data stream is applied to the USRT and to a 8-bit shift register for delay. If there is no information being transmitted, the output of the shift register appears as transmit data. The receive data is processed by the USRT which recognizes the basic protocol functions of Flag, Address, and Go-Ahead. The USRT also generates the appropriate check polynomials and will generate a signal which indicates that the polynomial agrees or disagrees with the transmitted polynomial.

If the link message block is not direction to this station, no action is taken by the USRT and telemetry channel hardware. The infomration is simply passed through the shift register. The determination of this station addressing is accomplished in two ways. First, the USRT allows the selection of a single station address which is loaded into the USRT after it is powered up. Second, the USRT will receive all messages and the Receive Here Memory Table selects those which will be

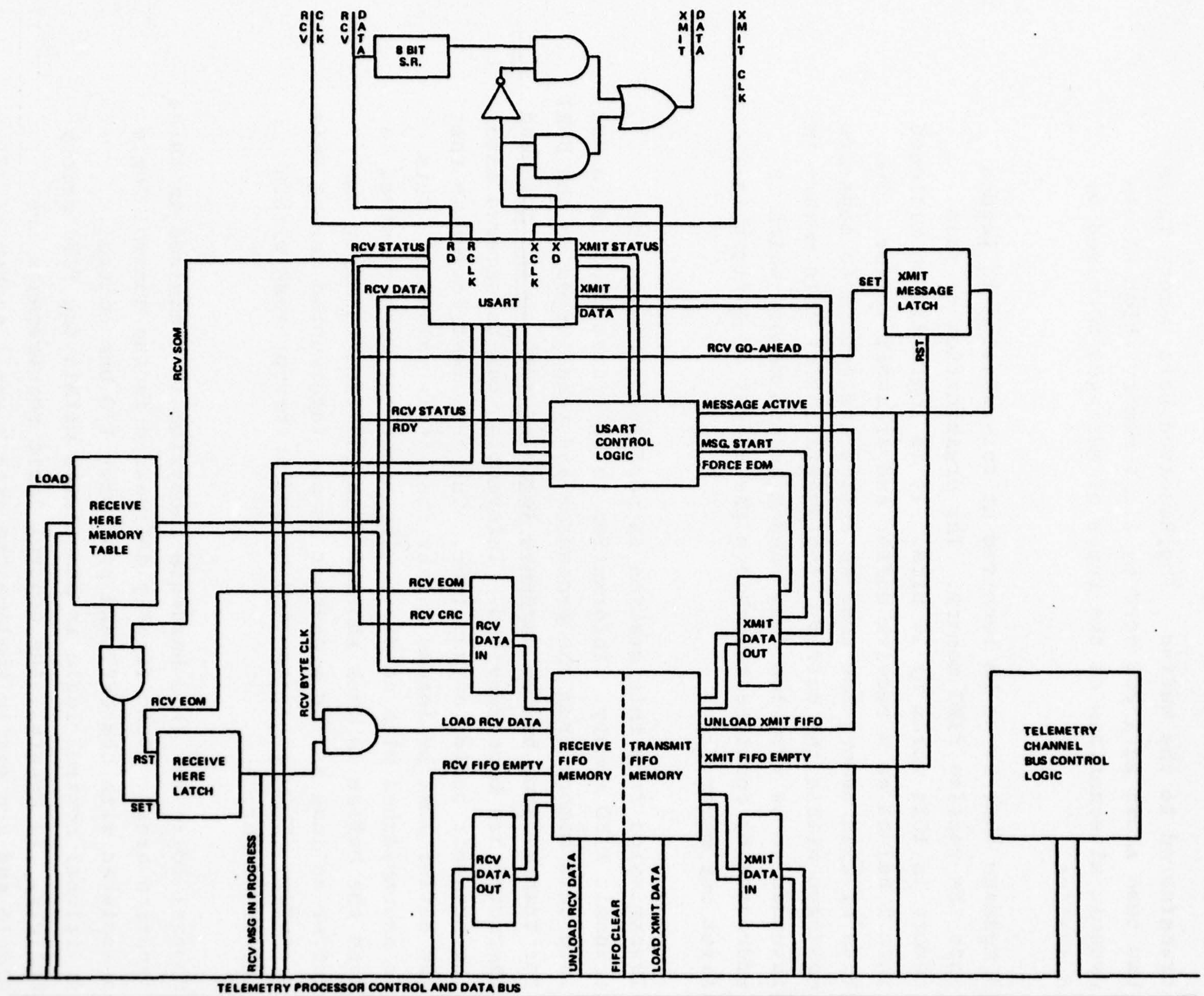


FIGURE 11-8
TELEMETRY CHANNEL HARDWARE

transferred to the buffer. The Received Here Memory Table has been shown as a 256 word by 1 bit memory which allows dynamic alternations of the types of messages received by a station.

A message that is to be received at this station is loaded into the receive FIFO memory. The organization of this memory is 1024 words by 10 bits. It is logically partitioned into 2 halves as a receive buffer and transmit buffer. Two bits of this memory are used as status indicators to separate messages within the buffer. Implementation of this memory is intended to be very low cost random access memory which is addressed by counters to achieve the effect of a first in-first out memory.

Transmission from this station is performed through the transmit FIFO memory. Information to be transmitted is loaded into this memory from the processor and transferred to the USRT for transmission by the hardware logic as soon as the go-ahead character has been received. Information can be entered into the transmit buffer at any time. The only restriction is that the buffer must be loaded faster than it is unloaded. This is accomplished with no difficulty since the time required to load the buffer is much shorter than the time to empty the buffer as long as the processor is not interrupted during this transfer. This is controlled by the interrupt mask within the processor.

Control logic for this hardware subsystem is contained in three separate areas. Specifically delineated is the control logic associated with the USRT and processor I/O bus control. Additional control logic is associated within the FIFO memory buffers. In general, the control logic requirements are simple and are easily implemented with a small number of MSI or SSI packages.

Although a large number of the required protocol functions are performed by the USRT , there is a set of protocol functions that must still be performed by the systems software. These include the intraloop message functions of frame sequencing, mode change detection/control, initialization, ACK/NAK dialogue and error recovery. System software also performs message routing and all of the protocol functions associated with interloop communications.

11.1.5 Data Acquisition Subsystem

After reviewing the functional requirements for the data acquisition system and the DEB network, it appears that there is no single, simple configuration which will satisfy all the needs of the network and maintain the low cost objectives. A data acquisition system optimized for a major station such as Donnersburg or Feldberg is overly complex for a simple RTU and vice-versa. The first major partition then becomes a partition based on station complexity. The amount of equipment at a station is, of course, the primary factor impacting the data acquisition hardware. For a station with three branches, the total equipment at this station cannot exceed 60 pieces (3 radio sets, 3 KG-81's, 6 level 2 MUX, 48 level 1 MUX) and invariably is much less than that.

It appears that, as a rough average, about half of the level 1 MUX capacity at a station is typically used, the other ports being unused or through-grouped. Using this rule-of-thumb, most stations with 3 or less branches have less than 50 pieces of equipment. If the network is partitioned into simple and complex stations using the 3-branch criteria as the dividing line, there are 99 simple and 11 complex stations.

Ignoring for the moment problems associated with level conversion, there is additional commonality between the various pieces of equipment within a station. Level 1 multiplexers, level 2 multiplexers, and KG-81 TEDs can be handled by a standard module which has 1 or 2 analog input points, 32 digital input points, and 5 digital output points. A radio set/service channel mux combination has 8 analog points and at least 38 digital input points. The control requirements of a radio set are satisfied by 5 digital output points.

This suggests a standard data acquisition module that would contain 4 analog input points, 28 digital input points, and 8 digital output points. Two of these modules would be required for a radio set, 1 module would be required for a level 2 multiplexer, half a module for a level 1 multiplexer, and quarter of a module for a KG-81.

An alternative partitioning of data acquisition modules would be to provide separate cards for digital points, analog points and control points. The equipment oriented partitioning has several advantages. First, if a card fails, the failure is confined to a narrow set of equipment. Second, adding equipment at a station usually involves adding a single data acquisition card -- two in the case of the radio. Third, addressability of control points is simplified.

A block diagram of the proposed standard module is shown in Figure 11-9. Given the very low cost of the hardware after the level conversion, very little is lost by potential unused points as long as the unused points are terminated to a known logic level.

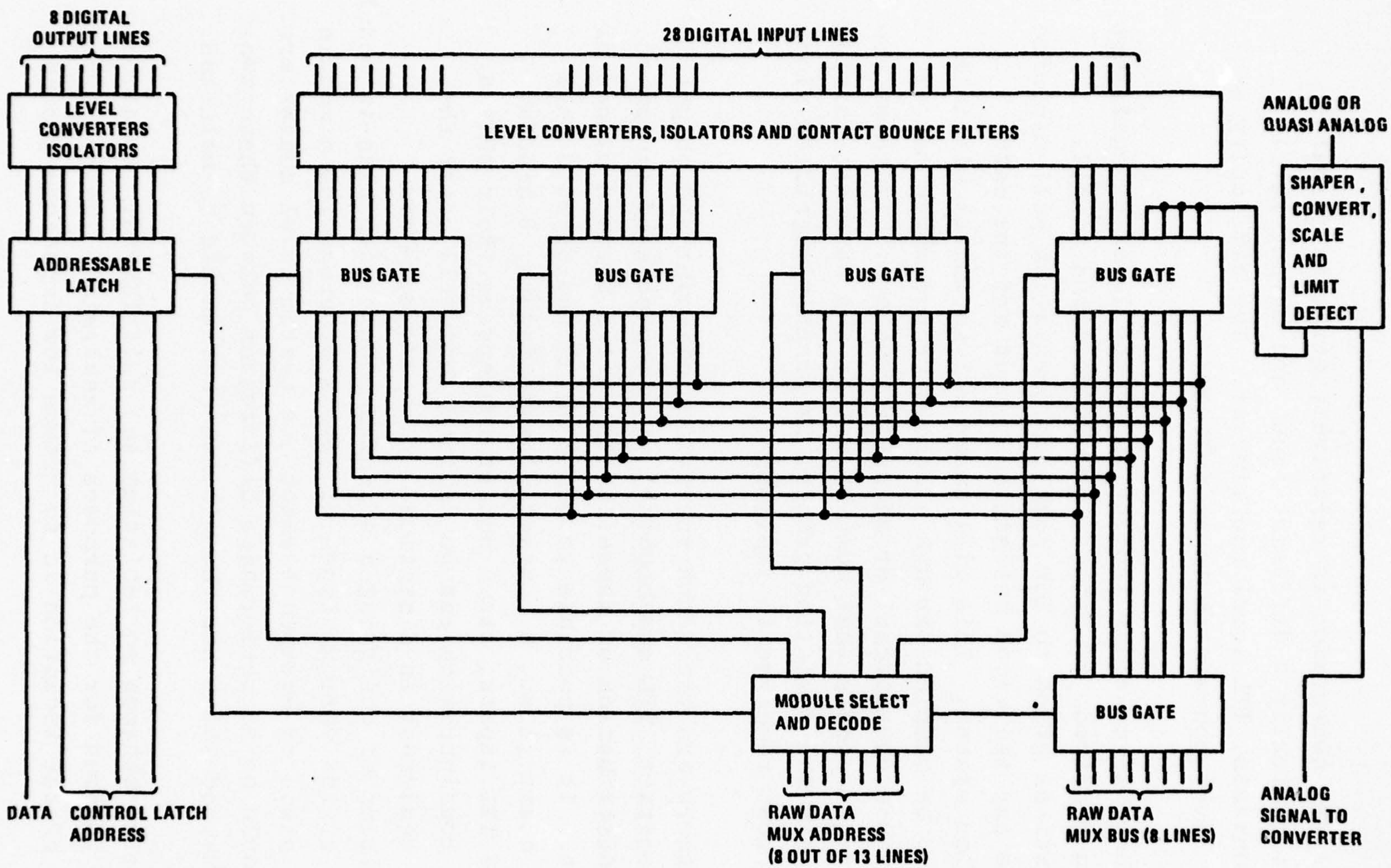


FIGURE 11-9
GENERAL PURPOSE DATA ACQUISITION MODULE

Digital level conversion is relatively straight forward. There are basically 3 different input and output levels: form C contacts, TTL level and bipolar. The difficulty in handling level conversion is as follows.

First, there is no clear indication of the level of equipment isolation required, if indeed, any is required at all. Considerations given to low cost implementation will minimize or delete any isolation between equipment and the data acquisition system. This will cause all equipment within a station to be coupled through a common ground which may or may not cause operational or security problems. If there are TEMPEST requirements that must be met by the data acquisition system, then the isolation problem is greatly magnified along with the cost required to implement.

Second, there are the known variations of input and output levels (contact, TTL and bipolar) within pieces of equipment and the distribution of these levels varies between pieces of equipment. It is possible to group inputs and outputs on a standard board (i.e., 8 contact closure inputs, 8 bipolar inputs 16 TTL inputs, etc.) but it is very unlikely that a standard configuration can be defined that will cover the range of equipment in a station. This suggests the possibilities of designing a standard module with plug-in modules for each of the various types of levels or producing a module for each piece of equipment which has custom level converters. There should be no performance differences between these two and the method which has the minimum cost should be selected.

A compromise between no isolation and full TEMPEST isolation has been assumed for the purposes of analysis. The primary function of the isolation is to reduce low impedance ground

loops to preserve isolation between branches. It is estimated that this level of isolation will required an average of .75 ICs per digital point.

Similar considerations exist for analog signal isolation. However, full D.C. isolation is a much more expensive proposition. A low cost optically coupled amplifier costs $43.00 for the amplifier only. A low cost instrumentation amplifier costs on the order of $3.00 in small quantities but does not supply the same degree of isolation. Given the desire for low cost implementation, the instrumentation amplifier approach has been used for analysis. For the purposes of analysis, an estimated 5 ICs per analog point have been assumed. The functions for the analog inputs are to terminate the analog signal, perform level conversion and out of limit comparison for generation of alarms. In the case of frame error rate signals, a frequency to voltage conversion is also performed.

From the block diagram of the multiplexer and TED equipment interface module, approximately 12 ICs will be required for the addressable latch, bus gates and module select and decode. About 30 ICs will be required for level conversion and 18 ICs for the analog input, making a total of approximately 60 ICs for this module. Approximately 35 IC spaces will be required for discrete components, yielding a total of 95 IC for the module.

Standard packaging practices should allow this circuitry to be packaged with an average IC density of 1 IC per 1.5 sq. in. with little or no difficulty. This yields a printed circuit board of 153 sq. in. or a board of 11"x14". Packing density can be increased to reduce this size with some increases in difficulty of construction.

The interface module provides the primary equipment interface and level conversion and reduces the data path from 32 parallel lines to 4 groups of 8 lines. Determination of alarm conditions, analog to digital conversion, and data routing to the telemetry or station processor has been partitioned to a common control unit. Analog signal processing is discussed in a subsequent section.

There are a number of ways to implement alarm change scanning. The previously defined functional requirements tend to limit some choices and complex hardware schemes increase the cost of others. Some assumptions concerning the nature of the alarm data and hardware/software response to alarms have been made. The first assumption is that a valid alarm condition will exist for the duration of the scanning period. Alarm change conditions which are shorter than the worst case scan access time do not represent useful data. Second, very little intelligence is required within the data acquisition hardware. Interpretation of what actually represents an alarm and what to do with this information is outside the scope of the data acquisition hardware. Third, the average processing rate of alarm conditions is faster than the rate at which alarms can occur.

Three alternative scanning methods were considered. First, use of available processor resources at a station. This was rejected. Previous work on the telemetry system indicates that a substantial amount of processing resources are required to perform the seemingly simple scanning task. If we assume a software routine which could perform the scanning task in as little as 1 instruction per input and 50% of the available resources to be used in this task of data acquisition as derived from the functional specifications, an instruction rate of 2 million instructions per sec is required. This is an unreasonable rate even for a minicomputer.

Second, a dedicated processing system could be included as part of the data acquisition system. This would require a processing rate of 1 million instructions per second. This rate is not achievable with a MOS microprocessor but it is well within the range of bipolar processing elements such as the Signetics 8x300 Interpreter. This very high speed processor can be implemented with 15 to 20 MSI and LSI parts and would meet the scanning time specification.

Last, a purely hardware scanning subsystem was considered. The hardware scanner addresses each digital input point from each piece of equipment and determines alarm change conditions. The location of each alarm change is placed in a small FIFO memory. This hardware scanner can be implemented with about 6 SSI, 10 MSI and 1 LSI parts. The scanning rate should be on the order of 300 nsec per point. The hardware control required for the scanning is shown in Figures 11-10 and 11-11.

Because of its overall simplicity, the hardware scanner is recommended. The scanning circuitry operates as follows. The scan counter will address an interface module and select a bit via the 8 to 1 MUX. This bit of raw data is compared to the previous value which is stored in the 4096 bit RAM. If the new value is different from the old value, the contents of the scan counter and the current data bit are entered into the alarm data buffer. A non-empty buffer will interrupt the attached processor. The new data bit is written into the random acess memory for use on the next scan and the scan counter is incremented to address the next input point.

Provisions have been made to stop the scanning operation under two conditions. The first condition is the alarm data buffer is filled. Continuing to scan and enter alarm changes with the buffer full causes loss of alarm changes and represents a fault condition. The second condition is requests for raw data

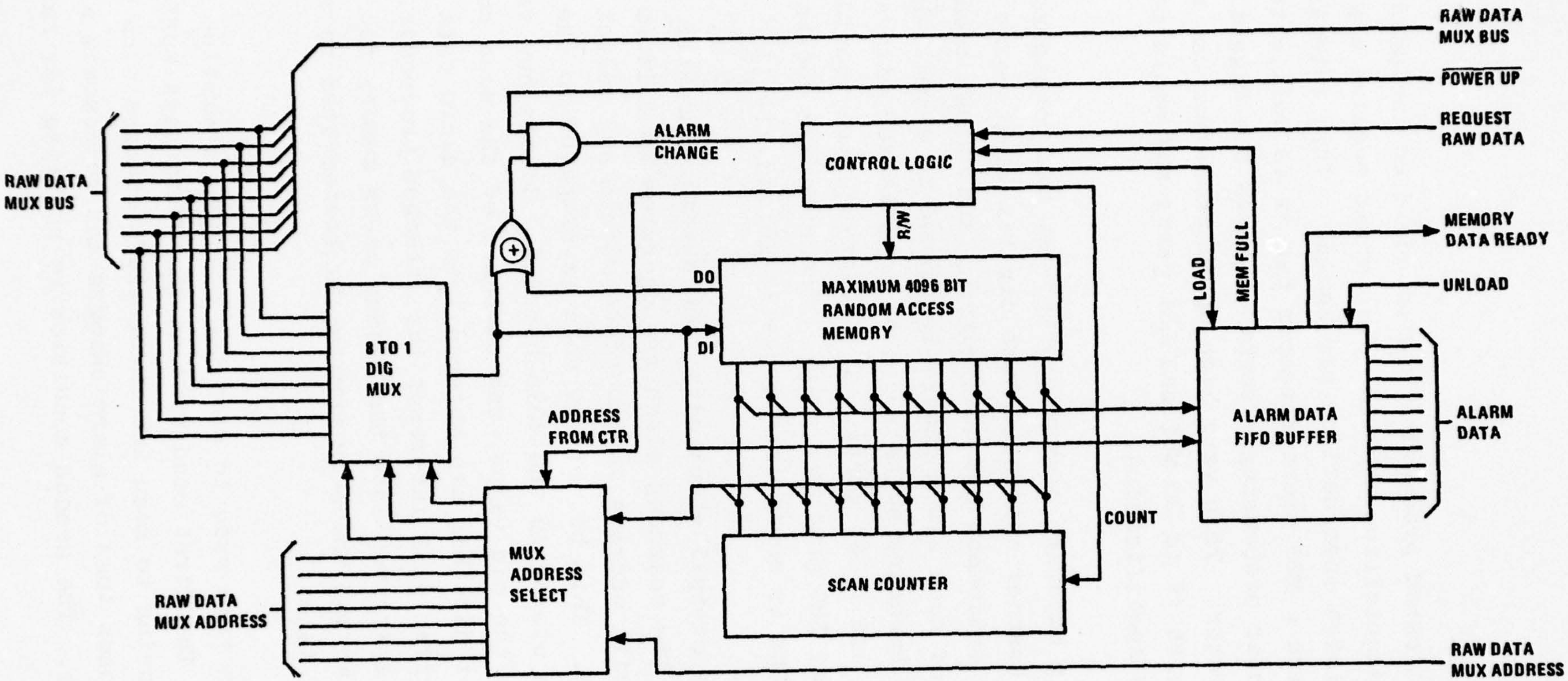
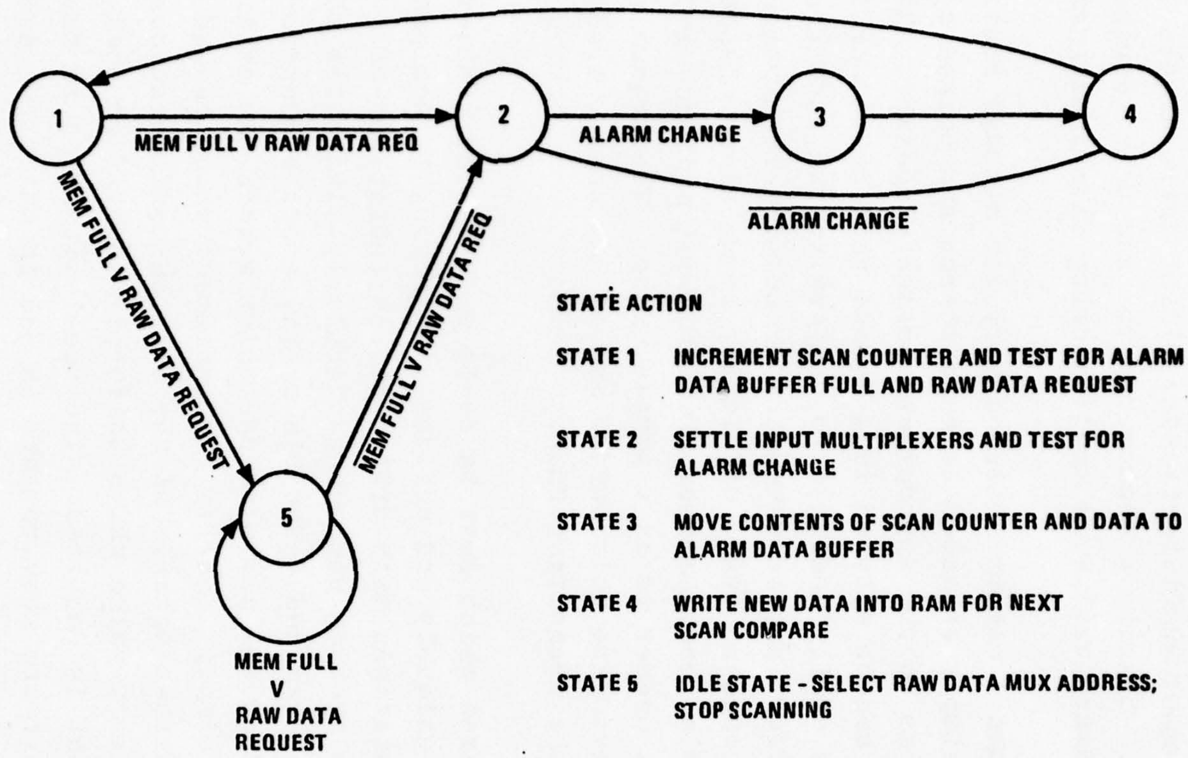


FIGURE 11-10
HARDWARE CONTROL FOR DATA SCANNING



STATE	ACTION
STATE 1	INCREMENT SCAN COUNTER AND TEST FOR ALARM DATA BUFFER FULL AND RAW DATA REQUEST
STATE 2	SETTLE INPUT MULTIPLEXERS AND TEST FOR ALARM CHANGE
STATE 3	MOVE CONTENTS OF SCAN COUNTER AND DATA TO ALARM DATA BUFFER
STATE 4	WRITE NEW DATA INTO RAM FOR NEXT SCAN COMPARE
STATE 5	IDLE STATE - SELECT RAW DATA MUX ADDRESS; STOP SCANNING

IF WE ALLOW 100 NS PER STATE, TO SCAN 1024 LOCATIONS WILL REQUIRE 150 μSEC.

FIGURE 11-11

DATA SCANNING CONTROL LOGIC STATE DIAGRAM

from the attached processor. A separate raw data bus can be utilized to gather equipment raw data as might be required for fault analysis and response to telemetry command. This would increase the complexity of the hardware both at the control unit and at the equipment interface for slight improvements in system response. Sharing a common bus for raw data requires the minimum hardware with some small penalty in scanning rate.

Having established a common shared memory for message routing, this memory is simply expanded to hold alarms and pointers to route these alarms to the appropriate branch processor. If this alarm data can be entered in a format that is either a link frame or is easily modified to conform to the link protocol, a substantial effort can be saved and processing requirements reduced. This poses no great difficulty if the alarm change buffer contains a pre-composed link protocol prologue and alarms are added to this header as in a simple queue. The branch processer must interrogate this buffer and if not empty, transfer this buffer to the transmit FIFO.

One more major area which must be dealt with is the problem associated with uniquely and uniformly referring to equipment groups between stations with different configurations of equipment. This problem becomes particularly clear when the scan counter is examined. The binary value of the counter uniquely identifies the alarm change point within the station but it does not simply identify the equipment which caused the alarm change. Either details of the station configurations or a standard binary value which uniformly identifies each piece of equipment is required. The ideal situation in terms of minimizing software requirements is one in which the equipment is identified in terms of branch, mission bit stream, and equipment. In order to perform manual and automatic fault isolation and restoral, this information will be required and must be generated and interpreted.

One simple method for accomplishing this identification is through a simple lookup conversion process. Shown in Figure 11-12 is a completely populated branch. If redundant equipment is considered as a pair of separate pieces or as two individual pieces of equipment, the greatest number of equipment that can exist at one branch is 24. This can be represented in five binary bits. From previous assumptions concerning the total number of branches that can exist at one station (16 branches), any equipment within a station can be specified with a 9-bit address (universal address).

Within a station, this 9-bit address must be converted to a module address which implies the maximum size table for this conversion will be 512 locations. A similar table is required for conversion from module address to universal address. This table is the size of the number of modules that actually exist within the station. In both cases the table size is a function of the actual station configuration rather than a standard, fixed allocation.

Some indication of the overall physical configuration of the system is presented in Figure 11-13. Both the circuit board and connector packaging density are well within the range of low cost commercial technology. Substantial cost savings can be realized through the use of commercial mass termination connectors both with the equipment connectors and card edge connectors. The suggested equipment connector is referred to as a "Micro Ribbon" connector and is extensively used by telephone companies for the multiple pair cables.

The anticipated packaging for the processor is in a second chassis which will also contain the power supplies for the TSC hardware. If the circuit board sizes of the processor are the same as the data acquisition, there is a possibility that, at many sites, the entire hardware required could be reasonably contained in a single chassis.

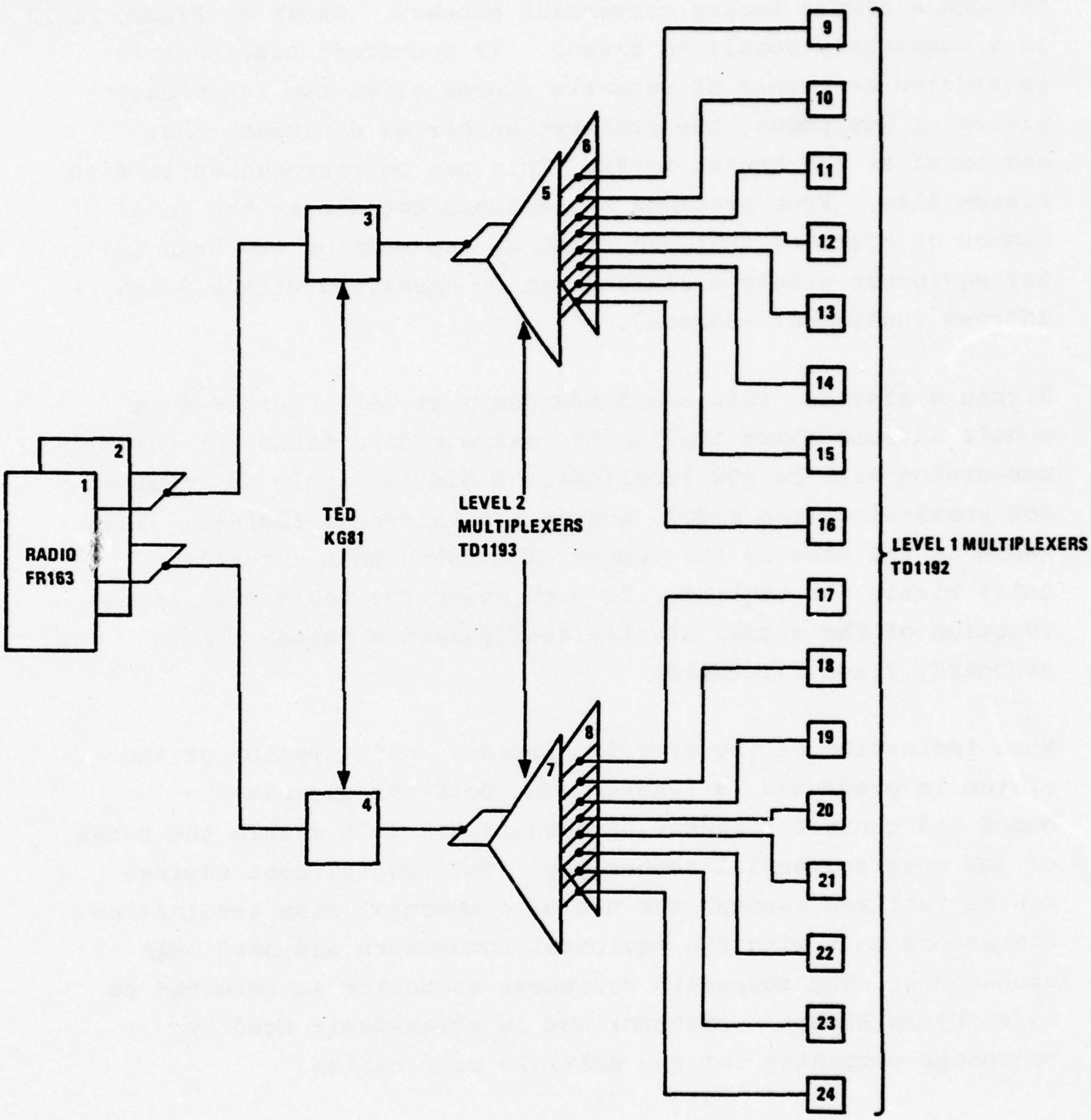


FIGURE 11-12
FULLY POPULATED BRANCH WITH EQUIPMENT ID NUMBERS

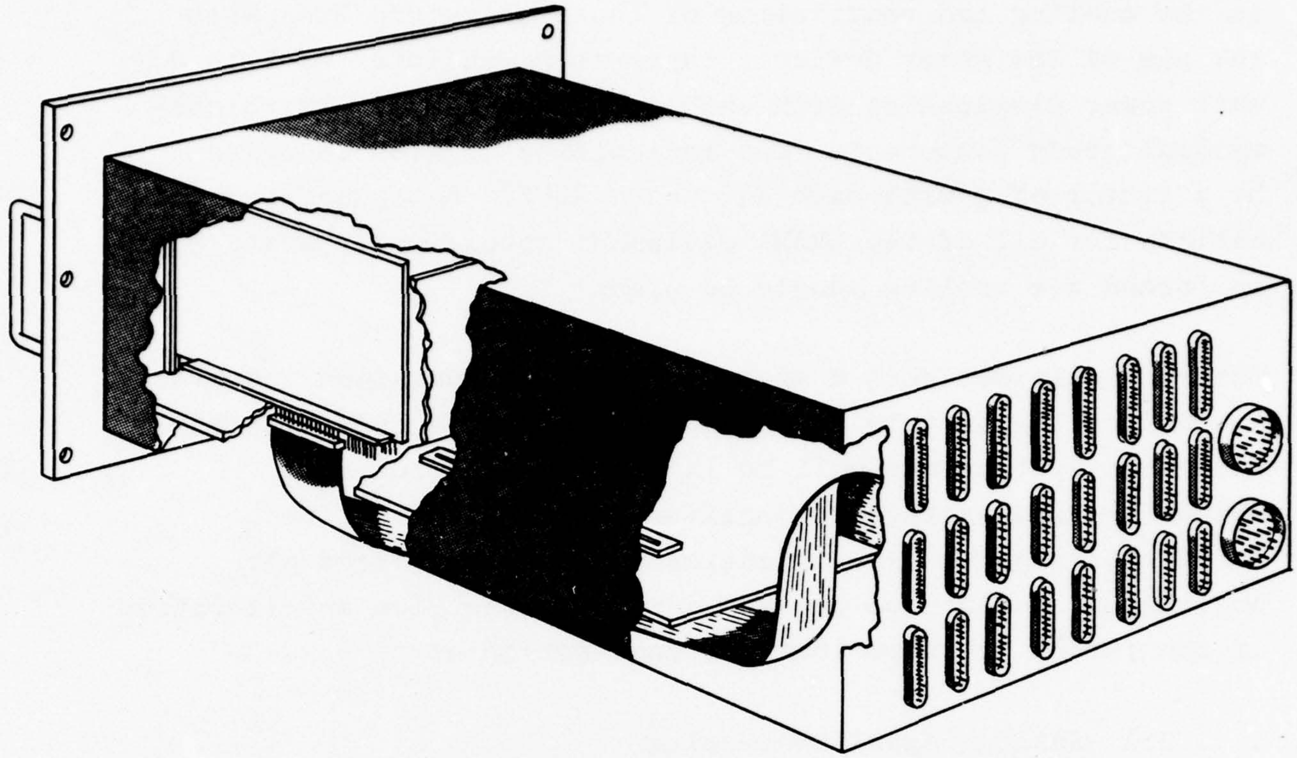


FIGURE 11-13
HARDWARE PACKAGING

The major problem to be considered with this packaging scheme is the cooling and ventilating of these chassis. Even with the use of low power devices, there is an anticipated 2 to 3 watt power dissipation with each interface module. With commercial grade components, the reliability or MTBF decreases by a factor of 2 with each 5°C above 25°C. Equipment specifications for all of the DRAMA equipment specifically state that no forced air cooling should be used.

The rationale for such a specification is understood, however, a modification to this specification for the acquisition and telemetry systems may well be in order if the low cost objectives and packagingobjectives are to be met. The modification to the specification would allow forced air cooling but would also provide a temperature rise specification if the forced air were lost for some period of time.

11.1.5.1 Analog Signal Processing

Two classes of signals, which can potentially be handled through analog signal processing techniques, can be identified within the DRAMA equipment. The first class of signals represents true continuously variable signals such as power supply voltage, received signal level, transmitter power level, etc. The second class of signals is presented to the data acquisition system in a digital (pulse) format and can potentially be treated either by digital or analog processing techniques. These signals come from the bit error rate (BER) circuitry of the radio, Level 2 MUX and Level 1 MUX and will be referred to as quasi-analog signals.

Primary power alarms and power supply alarms are specified with each piece of DRAMA equipment. It is assumed that the power supply alarms are window-type alarms and that they accurately monitor the power supplies over the specified operating range of the equipment.

Signal quality monitors are provided for both the on-line and standby receivers. The dynamic range is spedified as 40 dB with no sensitivity given. It has been assumed that the sensitivity is similar to that of the received signal level and that the alarm point of 5 dB for a BER of $1x10^{-2}$ represents the desired response.

Transmitter power and transmitter frequency drift have alarms associated wtih limits of range. Analog signals associated with these two alarms are probably not appropriate for telemetry monitoring although this can be accomodated if these are deemed useful to the tech controller.

Bit error rate signals are pulses which correspond to detected errors in the framing bits. The duration of the pulse is a function of the bit rate of the equipment. Alarm BER is a function of the equipment, ranging from $1x10^{-4}$ specified for the radio to $1x10^{-2}$ anticipated for the Level 1 and Level 2 multiplexers (the value has not been specified). Two needs for BER measurements exist: First, an alarm when BER exceeds a defined rate; second, as a continuous variable to monitor system performance.

The measurement of BER can be accomplished through either digital or analog techniques. Digitally, BER can be measured by counting over a known time interval and converting the result to a rate. An alternative is an

analog technique employing a frequency to voltage (F/V) converter. F/V converters, which have a dyanmic range of 10^4, are available as monolithic ICs. Converters with a dynamic range of 10^6 are available as prepackaged modules.

A minimum analog BER measurement circuit (excluding alarm comparator internal to radio) can be done with a single IC and about eight small discrete parts. Accuracy of the circuit is a function of the error rate and becomes more accurate as the error rate increases. A digital BER measurement circuit can be built with about six SSI and MSI parts. Alarm comparison is substantially more difficult digitally, given that the threshold must change between equipment and response time is limited by the dynamic range. The suggested implementation employs a F/V converter and analog comparator for alarm generation.

Measurement of power supply voltages and transmitter output power are not suggested. The DRAMA equipment provides out-of-range alarms for these signals and the benefits of providing telemetry access to these signals are marginal in both automatic and manual fault isolation.

Overall accuracy requirements for analog signal measurment are not high. Received signal level and signal quality monitor are adequately measured to within 1 dB. BER measurements are desired only to the exponent of the error rate. Both of these requirements can be satisfied with a total resolution of six bits (64 levels of quantitization). An eight bit resolution represents a very low end for commercial practice. Therefore, analog-to-digital conversion with eight bits will be used for the analysis.

Of the multitude of analog-to-digital conversion methods, three appear to be appropriate to this sytem. First, a tracking A/D converter wtih a converter associated with each analog point. Second, a dual slope A/D converter either with a converter associated with each analog point or a single common converter using analog multiplexers. Third, a high speed successive approximation converter using analog multiplexers.

A tracking A/D converter can be implemented with as few as seven ICs per channel and provides more than adequate performance. The tracking converter is a continuous converter for signals that are wtihin its tracking range and, thus, involves no delay in obtaining a conversion once it has locked on the input signal. The converter consists of an up/down counter, digital-to-analog converter, and analog comparator. The input voltage is compared to the output of the digital-to-analog converter (DAC). If the input voltage is greater thatn the DAC output, the counter is incremented, if it is less, the counter is decremented. When the input voltage is wtihin 1/2 bit of the DAC output, the couner will "dither", alternating counting up 1 and down 1.

Monolithis ICs exist to simply implement a dual slope A/D converter. If a converter is allocated to each channel, about five ICs per channel will be required. Using a single converter and multiplexing into it will require about seven ICs for the basic converter plus about 1/4 IC per input channel. Operation of this converter is on a discrete sampling basis. At the beginning of a sampling period, a capacitor is charged by a current source which is directly proportional to the input voltage. This capacitor is charged for a preset period of time ususally determined by digital counting techniques. At the end of this interval

the capacitor is discharged by a constant current and the time to discharge this capacitor to zero represents the input voltage. Typical conversion times for this conversion method are on the order of 50 msec.

Successive approximation A/D conversion is usually associated with packaged system A/D converters and has been characterized as providing the highest throughput with moderately expensive hardware. Currently available ICs allow construction of a high speed A/D converter with as few as six parts for the basic converter. Operation of this converter consists of clearing the conversion register and setting the most significant register bit. If the output voltage from the DAC is greater than the input voltage, the most significant bit is reset. The next bit is set and the output of the DAC is compared to the input voltage. If the output voltage from the DAC is greater than the input voltage, the bit is reset. The conversion time for this method is fixed by the number of bits required by the converter and can be performed in as little as 2 μsec for eight bits.

The anticipated use of the A/D subsytem wtihin the data acquisition system is on a demand basis. Specific requests for analog data will be made on an as-needed basis and no requirement for continuous sequential conversion can be determined. Typical requirements for analog data will be in response to telemetry commands for raw data and occasional requests as required for trending.

Requests for analog values either from telemetry commands or for trending data will be routed through the processor resources allocated to data acquisition. This means that the A/D subsystem requires software as well as hardware integration into the system. Three possible integration levels exist

within the bounds of the current processing system. These are immediate conversions, short delay conversion and long delay conversion. Immediate conversion requires that the data be available within one instruction time. Short delay conversion measn that the result of a request for A/D conversion is not available for some number of instruction times, but the number of instructions is not sufficient to perform other tasks during this period.

Long delay conversion requires a large number of instruction times such that useful work on other tasks could be performed while the conversion is in progress.

With immediate conversion, the processor is unaware that it is dealing with anything other than its normal I/O devices. It simply addresses the desired channel and results are available with the next instruction when the processor reads the device. This method is the simplest to integrate into the software, since analog I/O is no different than digital I/O. Fast conversion times are required and potentially increase the performance requirements of the A/D converter.

The boundary between short delay conversion and long delay conversion is fuzzy. It is related to the interrupt processing overhead and the difficulties involved in implementing an interrupt service routine. An approximate bound lies at 30 instruction times plus interrupt service overhead. With both short and long delay conversion, the processor must know that there is an A/D converter wtihin the system so that the proper routine is entered to wait for the converter delay. Short delay conversion is probably best handled with a software wait routine in which the processor is in a hard loop waiting

for the converter to finish. Long delay conversion lends itself to interrupt processing in which the processor initiates the conversion cycle and performs some other task while waiting for the converter to complete.

Immediate conversion is suggested for implementation for several reasons. First, it is the simplest to integrate into the system. No special software considerations are required. Second, for the targeted processor for this system, performance requirements placed upon the A/D conversion system are not extreme. Third, immediate conversion is the most tolerant of failures within the A/D subsystem. For certain converter failures, the immediate conversion method will allow the processor to continue. Fourth, immediate conversion provides the fastest throughput. No delays or overhead are associated with this method.

Given an approximately 5 μsec conversion requirement of A/D conversion as determined from the immediate conversion method, dual slope conversion cannot be used. Tracking A/D conversion is attractive because of its immediate availability of data and distributed conversion capacity which enhances the overall survivability of the A/D subsystem. However, there is a substantial increase in the total amount of hardware required which will increase the system cost. On this basis, successive approximation conversion with multiplexed analog channels is recommended. This is shown in Figure 11-14.

Analog multiplexing can occur on the A/D converter card or on the equipment interface card. Locating the analog multiplexer on the A/D converter card allows the use of multiple

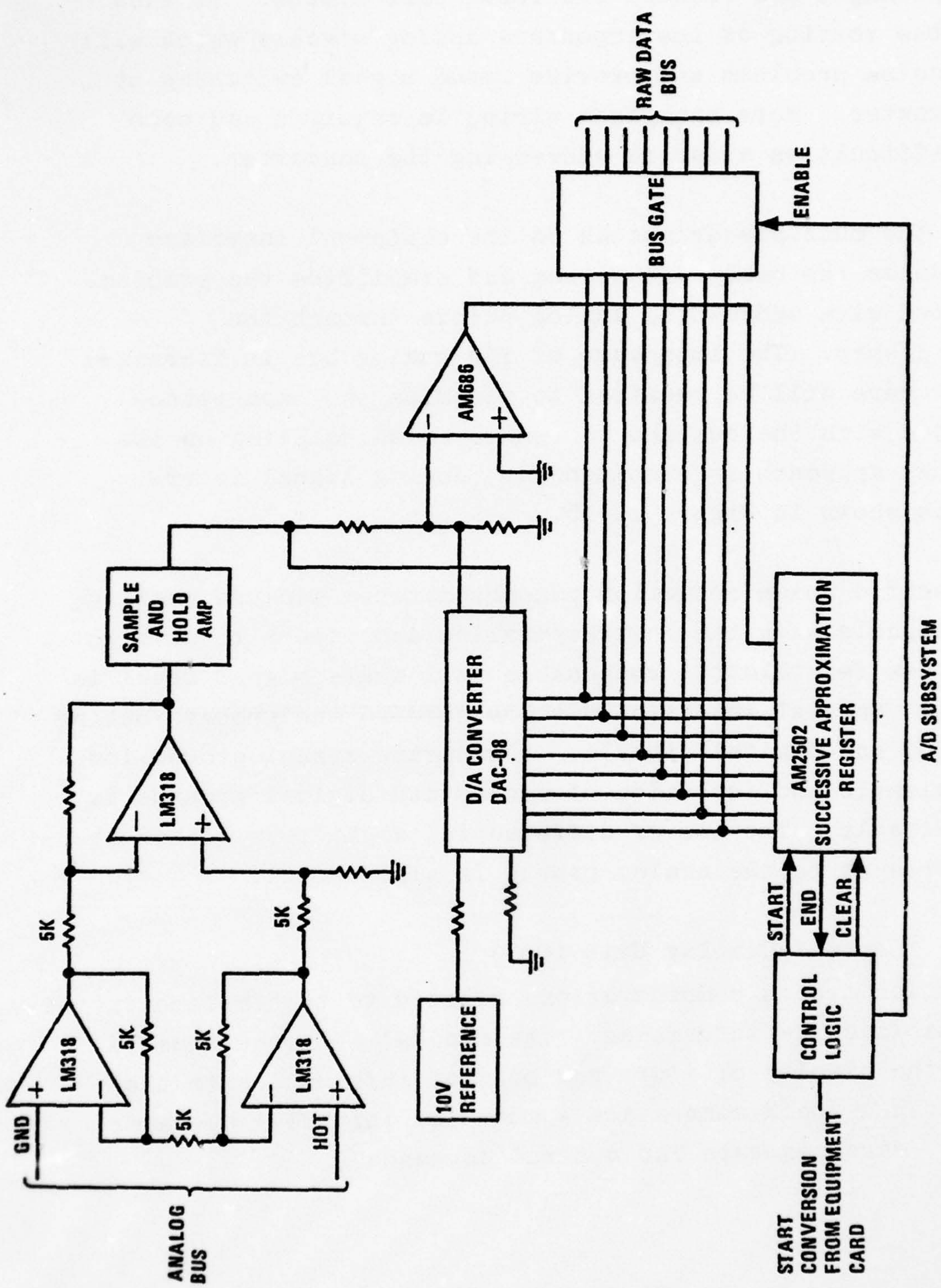


FIGURE 11-14
A/D SUBSYSTEM

switch packages and reduces the total part counts. It also allows the routing of low impedance analog signals which will reduce noise problems and provide rapid signal switching at the converter. More backplane wiring is required and some major difficulties exist in addressing the converter.

Placing the multiplexer switch on the equipment interface card reduces the backplane wiring and simplifies the problems associated with addressing analog points through the microprocessor. The impedance of the analog bus is increased and some care will be required to minimize the capacitance associated with the bus and to reduce noise coupling on the bus. This approach is recommended. Analog signal interfacing is shown in Figure 11-15.

Some specific noise reduction recommendations include scaling analog signals with the instrumentation amplifiers to as high as level as feasible. A reasonable full scale signal level is 10 volts. Special attention must be paid to the ground routing for analog and digital signals. A separate signal ground for analog signals and isolation of analog and digital grounds is very desirable. The use of differential amplifiers within the A/D referenced to the analog ground is also useful.

11.1.6 Control/Display Unit (CDU)

This section treats considerations related to the implementation of TSC man/machine interfaces. The minimal CDU requirements include the display of alarm and monitor information to the operator in a lucid manner and a means of inputting operator messages, data requests and control commands.

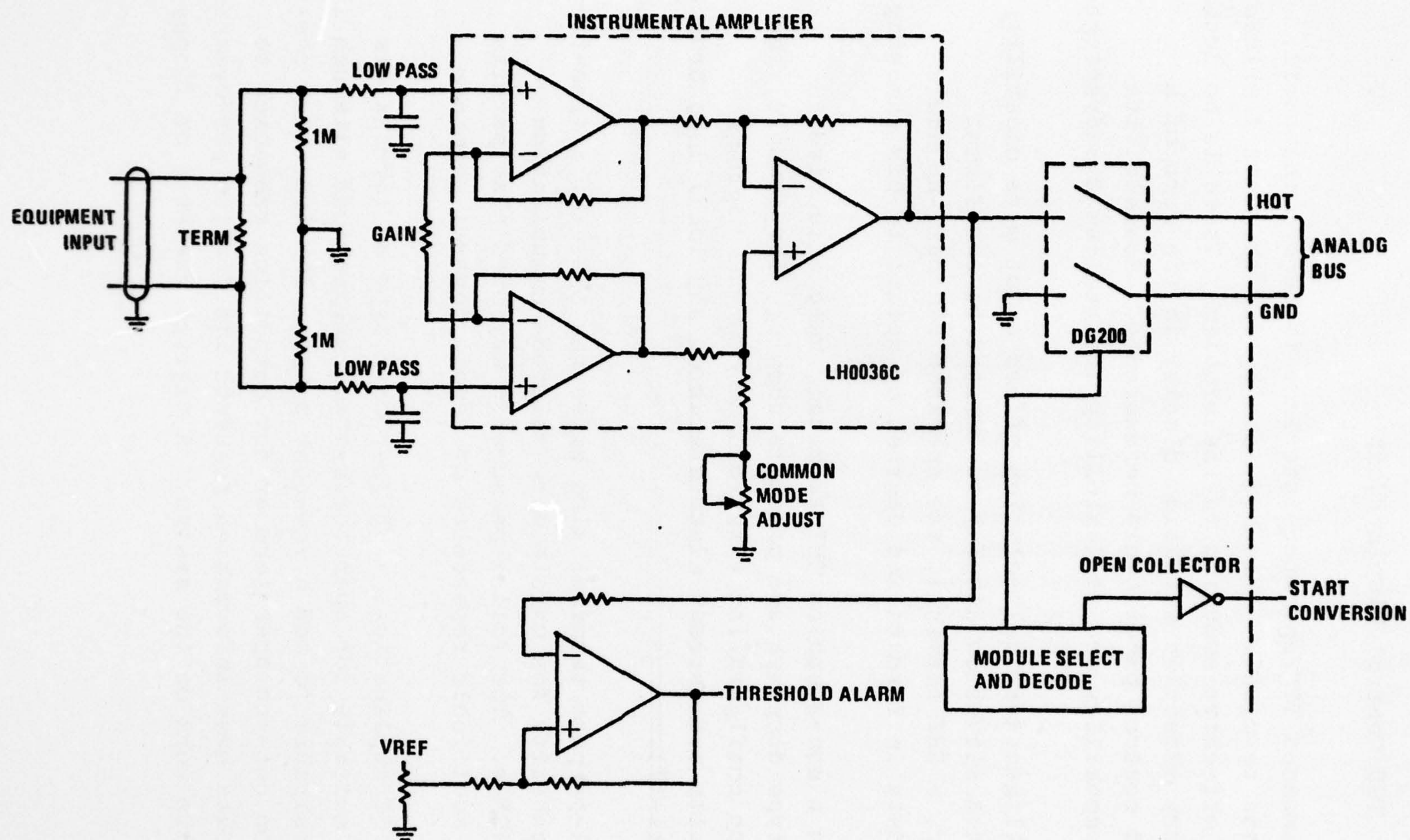


FIGURE 11-15

EQUIPMENT INTERFACE MODULE - ANALOG SIGNAL PROCESSING

11.1.6.1 TCU Control Display Unit

The recommended TCU/operator interface is an off-the-shelf, intelligent terminal. This provides by far the most flexible and cost effective solution to the problem. There is no doubt that a very effective situation display in the form of a dedicated control panel could be designed. However, its limited capability and inflexibility are serious disadvantages.

The intelligent terminal solution offers a lot more capability. A dedicated situation display is limited to one format. With, say, a CRT terminal, the operator can call-up and display data in formats and degrees of detail to his choosing.

Even with a non-graphics CRT terminal, very effective graphic-type displays are possible showing, for example, the local loop configuration (Figure 11-16) or the digroup connectivity and current alarm status on any local loop branch (Figure 11-17).

A general-purpose terminal also gives the operator a free-text input capability for maintenance-related communication between TCF's. The full-alphanumeric keyboard also permits a large, modifiable repertoire of remote control commands.

For the TSC application, a CRT/keyboard type of terminal is the most suitable for operator/system interactions although it may be desirable to add a low-cost printer for free-text communication between operators or for outputting responses to central data base information requests and for the performance of software work on the system. A digital cassette or floppy

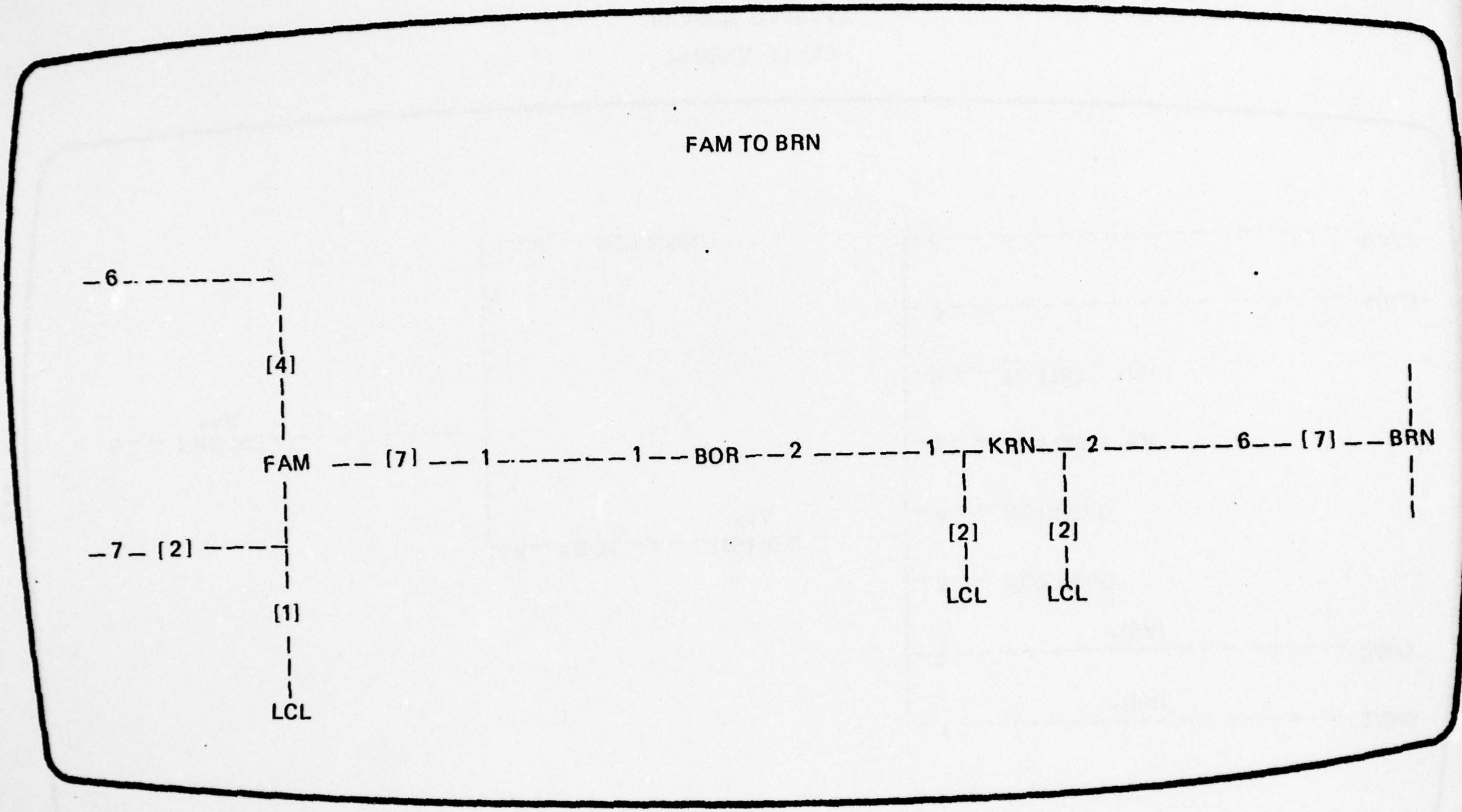


FIGURE 11-16
LOCAL LOOP DISPLAY

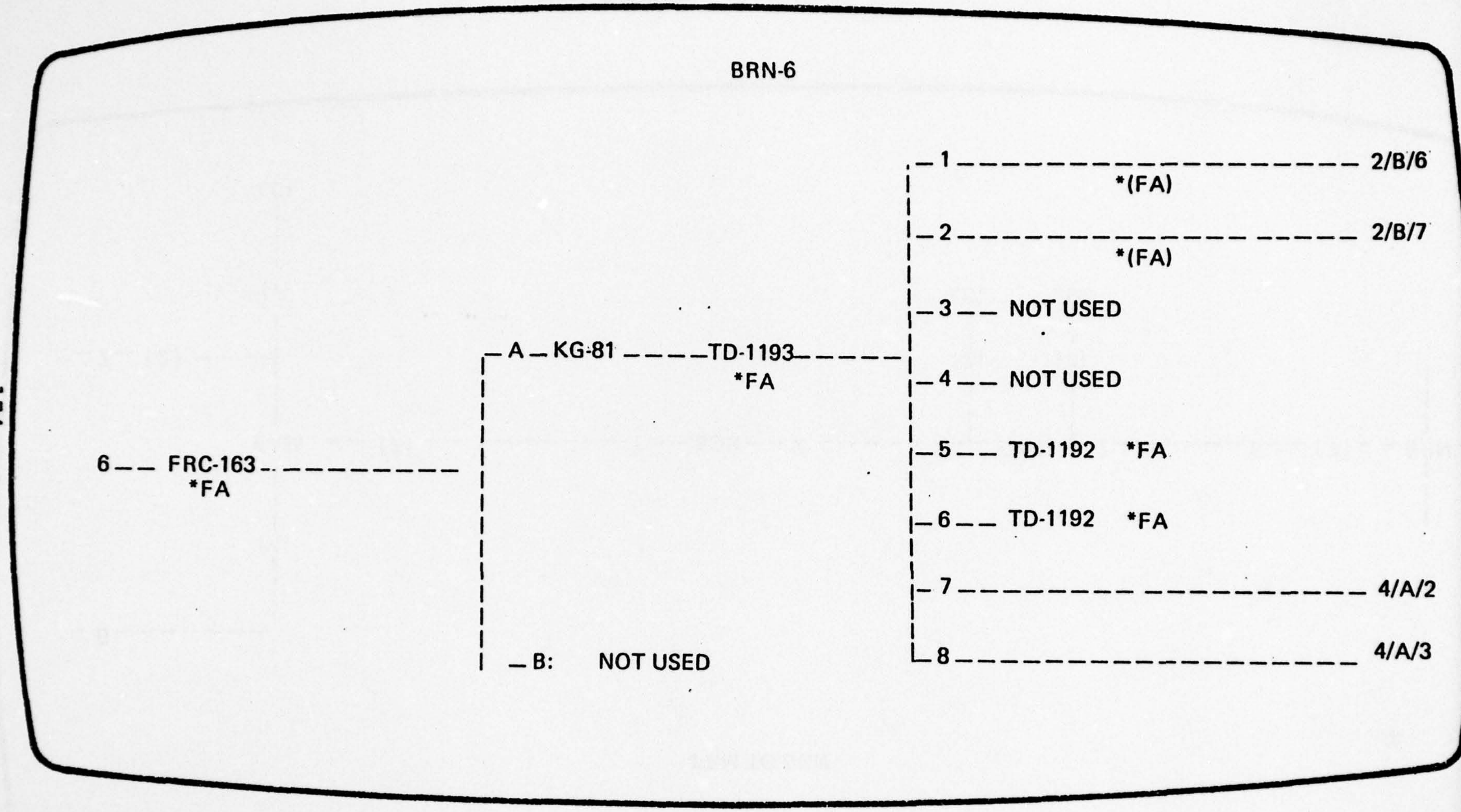


FIGURE 11-17
BRANCH DISPLAY

disk capability may be desirable for event logging and is essential for doing any on-site software modification and maintenance. Of course, for the latter need, a portable unit could be used.

It is envisioned that terminal firmware (probably UV ROM) would store terminal and operator assistance routines for functions such as data formatting, operator lead-through, entry validation, etc.

11.1.6.2 RTU Control/Display Unit

The situation at an RTU is different. RTU locations with VF drops are presumed to be manned sites and a CDU similar to that deployed at TCU locations is appropriate.

At unmanned repeater sites, however, the only time that the capabilities of a CDU are needed is when a maintenance team is present. The sensible and most economical solution, then, appears to be a portable intelligent terminal. With this approach, the RTU would be equipped with a CDU interface where a portable terminal could be plugged-in.

A state-of-the-art portable intelligent terminal can be programmed to provide valuable diagnostic aids to the maintenance teams. The TI Model 765 portable terminal, for example, can be purchased with up to 80,000 bytes of bubble memory! Bubble memory is, of course, non-volatile and 80,000 bytes can store a lot of sophisticated diagnostics, technical data, etc.

It is envisioned that this portable terminal would give the roving maintenance team access to the local loop telemetry for the purposes of requesting specific alarm and monitor data, issuing remote control commands, and communicating with TCF's.

11.2 Software Implementation

11.2.1 General Organization

The TSC is composed of both hardware and software both of which are critical to the overall system performance. Having selected a system architecture, both hardware and software require careful integration. This is especially true with the recommended implementation of a single processor based TCU and RTU. For certain stations with a large number of branches and/or network control responsibilities, processing efficiency is very important if this implementation is to provide the desired performance.

Some general organizational requirements are immediately defined in keeping with current programming practices. These can be generally described under the areas of structured programming which control the construction of the software. Some specific considerations are top-down programming, localization of control, and program modularity.

Top-down programming begins with the system as a whole and decomposes the system into a series of functions which are decomposed repetitively until the machine executable object code of the processor is finally reached. Properly used, top-down structured programming can yield reliable software which is easily maintained independently of the programming language used.

One requirement of the structure of the software is a clear need to localize control. Specifically, separate software modules which are given exclusive control and manipulation capability are required. The need for this is twofold. First, since any specific control or manipulation occurs

at one clearly defined location within the software, any problems associated with this control and manipulation are localized. Second, any changes in control or manipulation that are desired are confined to one module and do not influence any others.

The recommended software architecture is task oriented. The tasks of the system provide the functional partitioning of the software as shown in Figure 11-18. Tasks are initiated by interrupts and by other tasks and can suspend their operation as they wait for additional information. Communication between the tasks is through a series of standard internal messages or parameters which are passed between tasks. In general, information maintained by one task is directly available to any other task. However, this information cannot be modified except by the responsible task.

An additional consideration in the construction of the TSC software, especially with reference to the functional partitioning between the tasks, is the desire to achieve as much commonality as possible between RTU and TCU software. Ideally, RTU software should be a simple subset of TCU software.

11.2.2 Program and Information Flow

Function partitioning between the tasks creates some potential difficulties in understanding the overall software operation if some care is not taken in recognizing where and how decisions are made and actions taken. Appendix C of this report presents the detailed recommended software implementation and only the more general functions will be described here.

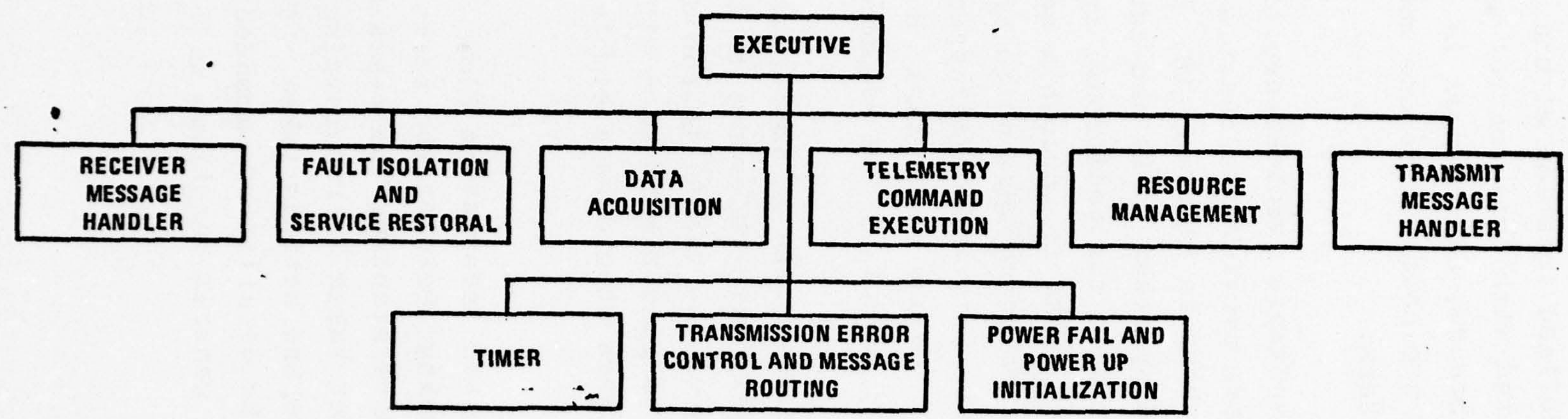


FIGURE 11-18

SOFTWARE ARCHITECTURE

Some of the key concepts, which are required for understanding have already been mentioned. First, each task has a definite range of responsibility and action. Frequently, this responsibility and action are only a small portion of the total required. Second, the tasks communicate between themselves with messages and/or parameters. Third, execution of a task can be suspended either by the occurrance of an interrupt from the hardware or by the task itself when it requires additional information. When a task is suspended, the processor is free to perform some other task.

Other key concepts are as follows. First, only one task can be actively executing at one time. This is a function of the single processor implementation. A large number of tasks can be in an active but suspended state at any given instant. Activation and reactivation is usually on a first-come, first-serve basis. The primary exception to this is the reactivation after servicing a hardware interrupt. In this case, the usual action is to reactivate the task that was executing immediately prior to the occurrance of the interrupt.

The major functions of the tasks for a TCU are as follows:

Executive: Maintains the list of tasks to be executed and handles the initiation and reactivation of tasks. The executive has primary responsibility of establishing interrupt priorities and coordination of communication between tasks. Diagnostic software is also part of the Executive.

Memory Management: Allocates memory areas to tasks as they require memory. Memory is allocated from a memory pool which the memory management task must maintain.

Timer: Software time interval generator which is required to schedule tasks and to reactivate them after predefined intervals have elapsed.

Power Failure and Power Up Initialization: Required to place the system in a known operational state with either of these conditions. A special entry point to this task will allow a partial reinitialization as may be required after long outages.

Receive Message Handler: Primary software interface between receive side telemetry channel hardware and TSC processor. Messages received via the telemetry channel are moved into the TSC memory and tested for errors. The receive message handler passes information to the transmission error control task and determines tasks to be activated on the basis of the message content.

Telemetry Command Execution: This task interprets each received telemetry command (including internally generated switching commands) and performs the specified operation. This task is responsible for maintaining all data base information which is subject to telemetry manipulation and is also the primary software interface to the control hardware of the data acquisition subsystem.

Transmission Error Control and Message Routing: Responsible for the ACK/NAK requirements of both the link and network protocols. Network messages other than status reporting messages are routed by this function.

Transmit Message Handler: Primary software interface between the transmit side of the telemetry channel hardware and TSC processor. This task performs the final message composition of protocol and information and places the complete message in the transmit buffer of the telemetry channel hardware.

Data Acquisition: This task is the software interface between the monitoring hardware of the data acquisition subsystem and the TSC processor. The task has an interrupt entry point for alarm changes and is responsible for determining the state of the data streams based upon local data acquisition information. This task generates messages to the operations personnel and information to the automatic fault isolation and restoral algorithm. The goodness of alarm information is determined by this task.

Automatic Fault Isolation and Service Restoral: This task is responsible for maintaining the state of the streams within a local loop. It performs alarm correlation, message generation, and equipment switching required for restoral and reports summary results to operations personnel.

As can be seen from this functional description of the tasks, virtually any stimulus to a TCU will involve several tasks. Some of the task involvements may appear to be overly complex. This complexity is needed, however, to satisfy the structured design and localization of control requirements.

To show the flow of control within the software, two examples will be presented in some detail. The first example is a remote telemetry command for equipment switching and the second example is the processing of an alarm condition from

a Level 2 Multiplexer. These examples are slightly simplified for the sake of clarity. Action of the executive and resource management tasks along with activity from other polling are not shown. The processing is shown in Figures 11-19 & 11-20 as time line processing. It is not intended that these represent real values for time.

Telemetry command processing proceeds as follows (letter designations are from Figure 11-20):

A) Telemetry channel hardware interrupts the CPU and the receive message handler task is activated. This task moves the message into the processor memory and tests the block check for errors. The message is interpreted and control set up to the telemetry command processor.

B) The transmission error control task composes the required ACK response for this telemetry command message.

C) The telemetry command processor further interprets the message to determine the action required and the options to be used with this command.

D) Since this is a switching command, the state of the equipment must be determined so the data acquisition task is initiated and the equipment raw data is obtained.

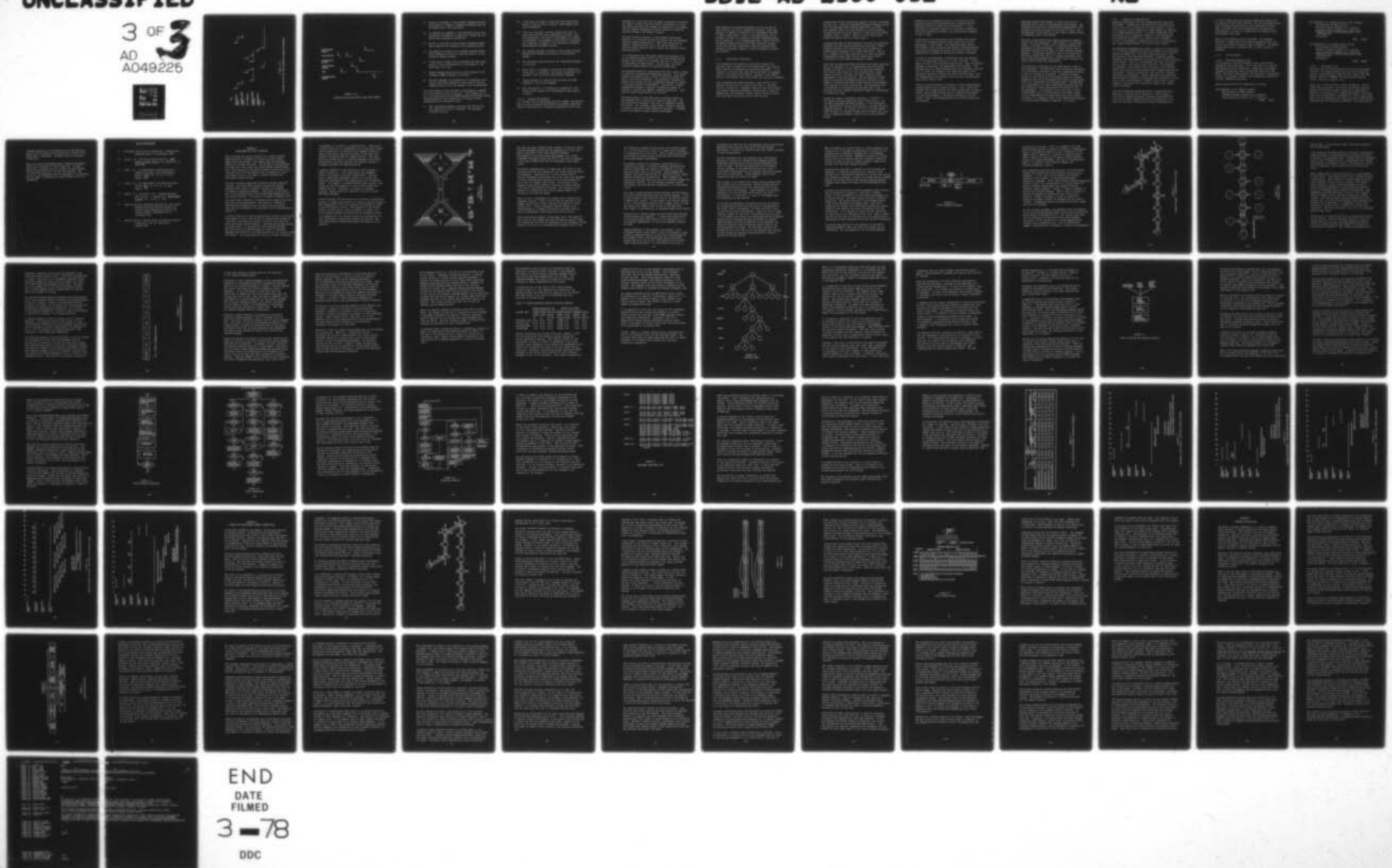
E) Assuming that the state of the equipment and the options for the command do not preclude switching. The switching is performed by the telemetry command processor.

F) After switching, a delay is introduced to allow the equipment status to change following this switching action.

AD-A049 225 E-SYSTEMS INC ST PETERSBURG FL ECI DIV F/G 17/2.1
TRANSMISSION SUBSYSTEM CONTROL ANALYSIS AND DEVELOPMENT.(U)
JUL 77 R K SMITH, J N BEAUCHAMP DCA100-76-C-0036

UNCLASSIFIED SBIE-AD-E100 012 NL

3 OF 3
AD
A049225



END
DATE
FILMED
3-78
DDC


OF 3
9225

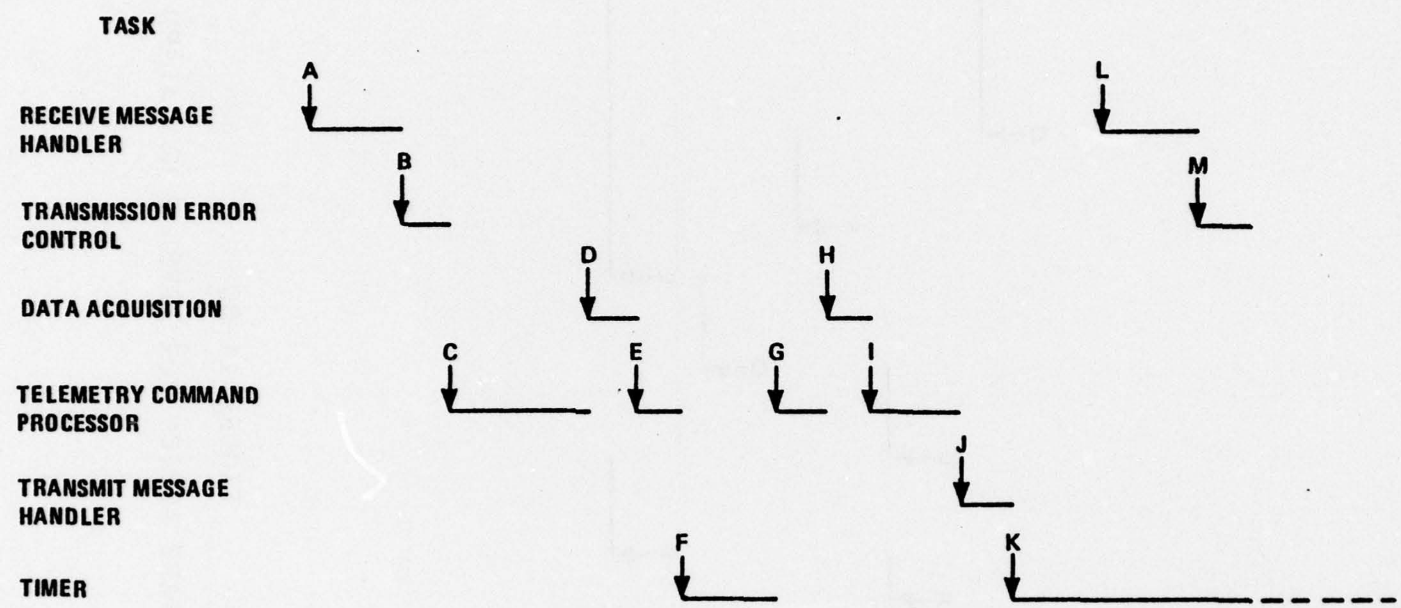


FIGURE 11-19

SOFTWARE ACTION FOR THE TELEMETRY COMMAND PROCESSING

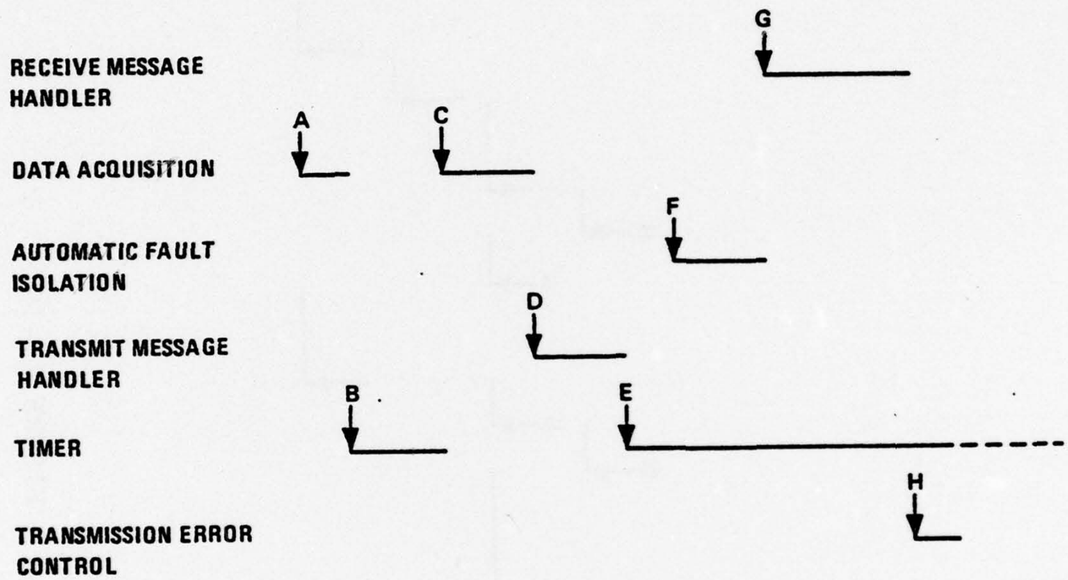


FIGURE 11-20

SOFTWARE ACTION FOR LEVEL 2 MUX ALARM CHANGE

G) Control is returned to the telemetry command processor which begins to compose a response confirming that the switching action has taken place.

H) To complete the response to the switching action, data acquisition information is required to verify that the action was successfully carried out.

I) Control is returned to the telemetry command processor which completes the response to the swtiching action.

J) This message is passed to the transmit message handler which composes the protocol frame and moves it to the telemetry channel hardware.

K) A time delay is begun after the message has been transmitted to allow retransmission of the message in the event that it is lost.

L) Another message which contains an ACK response to the telemetry command response is received.

M) This ACK response is processed by the tranmission error control task which removes the time interval from the schedule queue so that the message is not retransmitted.

The alarm condition from the level 2 multiplexer is assumed to be the failure of the on-line unit. The standby unit is in a not failed state and the DRAMA automatic switchover occurs. TSC processing for this condition is as follows (letter designations are from Figure 11-19):

A) Data acquisition hardware interrupts the CPU and the data acquisition task is initiated. The equipment involved is noted.

B) A time delay is begun to allow any other alarm conditions to occur so that a complete alarm change state can be determined.

C) After the time delay, the data acquisition task is reactivated and the alarm conditions are read. From this information, the state of the mission bit stream is determined, the goodness of the alarms is evaluated, and a message is composed to the operations personnel indicating the equipment failure.

D) The composed message is passed to the transmit message handler which proceed as outlined in the previous example.

E) The time delay associated with the transmitted message is activated.

F) Since there is a change in status for the mission bit-stream (OK to vulnerable), the status is passed to the automatic fault isolation and restoral algorithm.

G) Another message is received which contains the ACK response to the failure message.

H) This ACK response is processed by transmission error control which cancels the retransmission time interval as above.

11.2.3 Software Maintenance

In a network that is undergoing continual change, the requirement of ease of software maintenance is extremely important. Specific requirements for software maintenance occur in the

building of a local data base as might be required as stations are added or deleted from the network, modification to local data bases resulting from changes in connectivity or equipment, and the complete reloading of station software as might be required through a TSC hardware failure.

The most versatile solution to this software maintenance problem is to allow both on-site and remote (via the telemetry channel) software modification. Some restrictions may be required, especially with use of the remote capacity, to maintain system integrity and to control the sources of software modification.

It is anticipated that the normal operations personnel will not be computer systems experts and that maintaining the TSC hardware and software is a secondary function to their goal of maintaining the DEB network. On this basis, the most desirable system requires the absolute minimum skill.

One promising method for approaching this ideal system involves an interactive building and generating function. This system presents the operator with a series of questions which are used to build the information. The operator's responses are checked for accuracy and the information formatted to conform to the system software requirements. This method is highly recommended although it requires a substantial amount of software to provide the simplest operator interaction. This function would be partitioned into the CDU.

Some modification to local data base is already implied in the automatic fault isolation and restoral algorithm. This was specifically confined to the state of equipment (failed/ not failed). As a minimum, simple remote telemetry commands to modify connectivity tables should be available. A maximum would allow complete remote software maintenance.

. Both hardware and software maintenance of the TSC can be facilitated by the use of diagnostic software. Since the purpose of diagnostic software is to troubleshoot the TSC, it doesn't make sense to rely on the downloading of this software (if there is a defect in the TSC, downloading may not be possible). It is recommended that a copy of the full diagnostic software be stored at TCUs (possibly resident in the CDU) and that maintenance teams be equipped with a portable intelligent terminal in which diagnostic routines are resident.

11.3 Performance Limitations

The recommended TSC system has sufficient hardware and software scope to handle the anticipated worst case station which contains 16 branches and associated equipment. It is important to realize that each branch and piece of equipment places a load upon the TSC system. Because of the dynamic nature of the system, each load divides the available resources.

After examining the system carefully, some factors can be removed from the list of limits on the system. The most obvious of these are: equipment loading on the data acquisition system; operations personnel interactions in terms of demands upon the telemetry channel; and all other functions that have been partitioned into the CDU function.

Loading upon the data acquisition hardware involves increasing the scan time. Given the recommended hardware scanning system with its very high speed scanning capability, the worst case response time to an alarm change in a fully populated 16 branch station is on the order of 1.5 msec with an average response time of .75 msec. Relative to the other delays within the system, this is a minor factor.

Loading placed upon the TSC processor from both the operations personnel and from the CDU function are also minor factors primarily because of the anticipated low utilization or comparatively infrequent use of these facilities. This is not to suggest that these functions do not load the system substantially during their use but rather their overall contribution to the system utilization is small.

Given what has been previously defined as the normal operating mode (all equipment in a no change state and only routine polling occuring on a local loop) and the high probability that even a large station will be involved with a single network failure at one time, the factor limiting system performance is the polling rate. The polling rate is determined by the system throughput which is a function of the number of branches which must be serviced and the processing time required to service each branch. The processing time is determined by the instruction execution rate of the processor and the software efficiency (number of instructions to perform the task).

During the course of processor evaluation, a portion of the receive message handler was coded in some detail. For the Texas Instrument TMS 9900 processor, it was determined that approximately 150 μsec was required to perform this poll handling function. To account for all of the other system tasks involved, an assumed 400 μsec will be used as the total time to completely process a routine poll on any arbitrary branch.

Assuming a zero propagation delay, the polling time that could be potentially maintained within a station with 16 branches would be 6.4 msec with no contention. If the allocation of processor resources to polling is reduced to 70% of the available resources, the polling time increases to about 9 msec.

Increasing the polling interval also increases the access time to the telemetry channel which reduces the overall performance of the system. The simulation model used in the automatic fault isolation and restoral algorihm analysis shows that the propagation delays for network traffic are substantially increased as the polling interval increases.

A review of the deployment of TCUs shows that the majority are at stations which contain 5 or fewer branches. In fact, there are only 3 stations which contain more: Bann (6 branches); Feldberg (8 branches); and Donnersberg (10 branches). The stations with 5 or fewer branches can easily maintain a polling rate of 4 msec (as assumed in the simulation model) with sufficient capacity to handle the other required tasks. From the assumptions of the service time of a poll, the processor duty cycle is 50% or less.

There are three alternative solutions to the processing speed problem posed by the 3 larger stations. First, normal TCU hardware can be deployed and the lower polling rate tolerated. Second, these larger stations can be logically partitioned into what would appear to be smaller stations to the TSC system. Third, the processing rate of the TSC processor can be increased.

Deploying standard TSC hardware at these sites has the advantage that no special considerations are required. The achievable polling rate at Donnersberg (assuming 50% of the processor resources allocated to polling) is 8 msec. The disadvantage of this solution is that these 3 large stations are important hubs in the network.

Logically, partitioning these large stations into smaller stations for the deployment of TSC hardware is conceptually simple. To divide a station into 2 stations requries the deployment of 2 TCU processors. A pseudo control loop is maintained between the two TCU's by extra telemetry channel hardware. The primary difficulty lies in the connectivity that occurs within the station. Software associated with status reporting messages and automatic fault isolation and restoral may require some modification.

Increasing the processing rate for these special cases is also reasonable. One additional factor which was considered in the processor selection is that the recommended processor is part of a family of processors. Currently available as part of this family is a minicomputer which is software compatible with the microprocessor and offers an approximate factor of 3 improvement in processing speed. Conversations with representative of Texas Instruments uncovered the fact that substantial speed improvements are planned for the microprocessor. In a period of one to two years, another microprocessor will be introduced into this family which will be approximately 2.5 times faster than their current offering. It is suggested that if this faster microprocessor is available when hardware is deployed, it be used.

11.4 Mechanical Implementation

Packaging requirements for the recommended TSC system are very modest. Assuming the same printed circuit board form factor (11" x 14") is used for the entire system, a total of 10 printed circuit boards and a power supply are required at a simple repeater station. At least 2 additional card locations will be required to support on site maintenance. An average TCU deployment at a station with 3 branches, 6 level 2 multiplexers, 6 KG-81s and 18 level 1 multiplexers will require approximately 35 printed circuit boards.

The mechanical packaging plan shown in Figure 11-13 partitions all of the data acquisition associated hardware to one chassis which will support up to 45 data acquisition cards. The TSC processor (and associated hardware), telemetry channel hardware, and power supplies are contained in a second similarly sized chassis. While it is recognized that many card slots are not used in the case of small stations, this packaging is suggested. The rationale for this choice is as follows: First, there are only two different basic chassis required for the system. This will reduce the total variety of hardware which must be maintained. Second, assuming a very generous distribution of hardware at each branch (an average of 1 radio set, 2 level 2 multiplexers, 3 KG-81s, and 6 level 1 multiplexers), the single data acquisition chassis will support about 5 branches.

From this mechanical packaging scheme, a simple RTU and a majority of stations which contain 5 or fewer branches, will have a standard equipment compliment consisting of 2 19"W x 16"H x 26"D chassis mounted in a standard 19" rack. From the available information of station configurations,

it is not likely that more than one additional chassis will be required at most other stations. For the anticipated worst case 16 branch station which is fully populated (16 radio sets, 32 KG-81s, 32 level 2 multiplexers, and 256 level 1 multiplexers), a total of 6 chassis will be required along with another relay rack.

Space for the CDU must also be provided. For unmanned stations, a simple 16"W x 24"L shelf incorporated into the relay rack at some convenient level is adequate. For manned stations which have a permanent CDU, a desk or table is suggested.

11.5 Cost Estimates

11.5.1 Recurring Hardware Costs

In deriving the recurring hardware costs, the following assumptions have been made. First, the entire 110 station DEB network would be implemented. Based upon our assumed equipment populations, this includes 255 radio sets, 380 KG81s, 380 level 2 multiplexers, and 750 level 1 multiplexers. Second, all hardware, software and documentation conform to the best commercial practices.

These recurring costs are summarized as follows:

RTU deployment at an unmanned repeater

- TSC processor and RTU software
- Telemetry channel hardware for 2 channels
- Data acquisition hardware for 2 radio sets

TOTAL $7000

RTU deployment at a manned drop and insert repeater

TSC processor and RTU software

Telemetry channel hardware for 2 channels

Data acquisition hardware for 2 radio sets, 4 KG81s, 4 level 2 multiplexers and 4 level 1 multiplexers

CDU display

TOTAL $14000

TCU deployment at an average manned station

TSC processor and TCU software

Telemetry channel hardware for 3 channels

Data acquisition hardware for 3 radio sets, 6 KG81s, 6 level 2 multiplexers and 18 level 1 multiplexers

CDU display

TOTAL $23500

11.5.2 Development Cost

Non-recurring costs are based upon an advanced development contract. No program elements are costs for reliability, quality, maintainability, human factors, safety, EMC, EMP, nuclear radiation, TEMPEST, and environmental testing (shock, vibration, humidity, etc.).

Total cost of this contract is about $1,260,000. This is broken down as follows. The initial development includes hardware and software design, production of appropriate manuals and fabrication of 2 engineering models. The first model will be permanently configured as a TCU and the second model will be configured to be used as either an RTU and TCU. Estmiated cost of this phase is $960,000. The second phase

includes production of 4 TCU ADM sets and 2 RTU ADM sets. These are to be installed and tested over a period of six months and Ft. Huachucha. Estimated cost of this phase is $300,000.

After the initial development phase, advanced development models can be produced in small quantities. In small quantities, recurring costs for an RTU are about $16,300 and recurring costs for a TCU are about $33,000. TCU costs will vary depending upon the equipment to be monitored and the number of branches over which telemetry is to be implemented.

List of References

1) E-Systems, ECI Division, Design Plan - Transmission Subsystem Control, 23 September 1976.

2) Pierce, J.R.,"How Far Can Data Loops Go?", IEEE Trans. on Comm. Theory, Vol. COM-20, No. 3, June 1972.

3) Coker, C.H., "An Experimental Interconnection of Computers through a Loop Transmission System," BSTJ, Vol. 51, No. 6, July-August, 1972.

4) Kropfl, W.J.,"An Experimental Data Block Switching System, BSTJ, Vol. 51, No. 6, July-August, 1972.

5) Donnan, R. A. and Kersey, J.R., "Synchronous Data Link Control: A Perspective," IBM Systems Journal, Vol. 13, No. 2, 1974

6) American National Standards Committee-X3, Sixth Draft Proposed American National Standard for Advanced Data Communication Control Procedures (ADCCP), X3534/589-Draft 6, 15 October 1976

7) IBM Corporation, Systems Network Architecture General Information, Pub. No. GA27-3102-0, January 1975.

APPENDIX A
ALGORITHMS FOR FAULT ISOLATION

Fault isolation and service restoral is a stream oriented algorithm which is designed to operate on three stream types within the network. The stream types are digroups, mission bit streams and links. The actions of the algorithm are based upon local loops that exist between network nodes. Local data acquisition and remote telemetry provide the local loop with all of the information that is required to allow intelligent action to be taken to isolate failures and potentially restore service.

The fault isolation algorithm is based on several important concepts. The first major concept involves the failure syndrome matrix which is based on circuits (or streams) as opposed to physical equipment. The second is the definition of the interactions of the streams among themselves and the stream hierarchy. The third is a method of merging local and remote data acquisition information into simple structures which are easily manipulated. The fourth is a method of distributing information about circuits over the network such that this information is available to all locations which require it.

A failure within the network precipitates a series of activities which begin as soon as the failure is detected and extend for a period of time past the restoral of service for both automatic and manual service restoral. This period has been called an episode and it is divided into three distinct periods. The first period is charactarized by a burst of telemetry messages which result in the identification of the failure within the network. The second period involves an orderly switching

of equipment in an effort to restore service. This period includes execution of a predefined list of swtiching actions and the generation of information and messages which will assist operators in repairing failures in the event that automatic switching does not restore service. The third period is another burst of telemetry messages that provide notification of service restoral and generate reports of equipment problems and algorithm action.

A general schematic of the hierarchy of data streams is shown in Figure A-1. This figure shows the equipment extents over which the various streams exist. The purpose of identifying equipment extents is to confine the algorithm consideration to an appropriate list of possible causes. For example, a hardware failure which causes a link outage can never be caused by a level 2 multiplexer port hardware failure. Similarly, an outage of a single, isolated digroup cannot be caused by a failure in radio TDM common hardware.

Stream hierarchy has been derived on the basis of containment Stream A is higher than Stream B if Stream A contains Stream B. This designation of level is required to establish an order of service restoral actions. The argument is very direct. Service restoral action, either automatic or manual, on a failed digroup problem (as defined by the equipment extents of a digroup) will not restore service on that digroup if a link stream which contains that digroup has failed. The link stream must be functioning before the digroup can be restored.

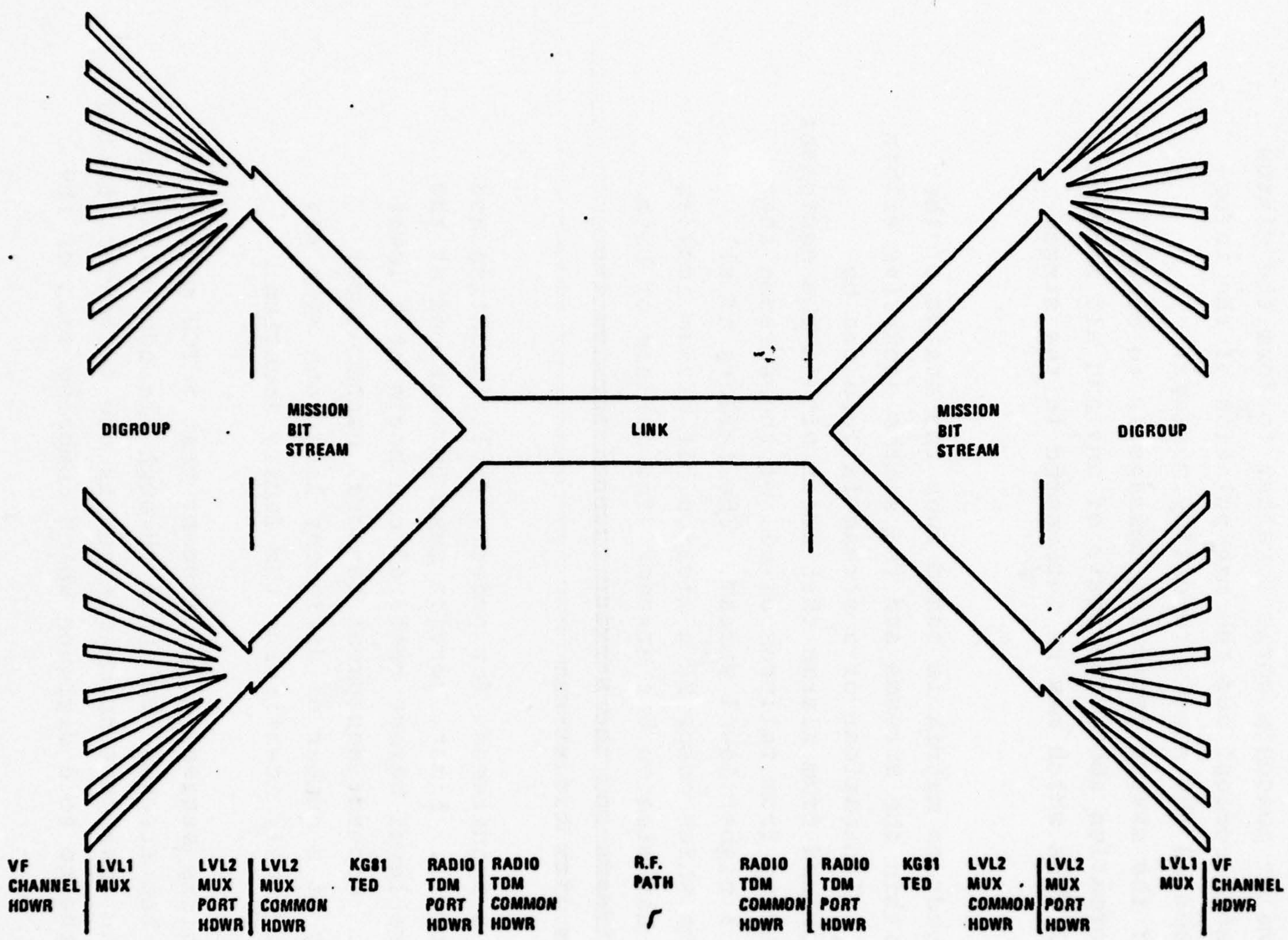


FIGURE A-1
STREAM HIERARCHY

The last two major concepts merge together to form the status reporting message concept and the manipulation of the information contained in the status reporting message.
The purpose of the status reporting message is to convey condensed information about the state of any and all streams to any and all TCUs which may be interested in the stream states.

The failure syndrome matrix is based upon the status of the information within the streams and the alarms occurring within the equipment. Indications of a stream failure can be directly determined from alarms that occur within the equipment or can be inferred from failures of all of the streams that are a part of a higher level stream. The failure of all digroup streams which comprise a mission bit stream implies a failure of that mission bit stream. The failure of both mission bit streams and the service channel implies the failure of the link bit stream

There is also a requirement for orderly fault isolation and equipment restoral. First, service must be restored at the highest failure level before restoral can begin at a lower failure level. Second, equipment restoral cannot occur simultaneously at a number of different locations with any hope of unambiguously identifying the faulty location.

These observations suggest a requirement that a TCU maintain the status of the streams that pass through its control loops. The information that must be maintained is the failed/not failed condition of the up to 8 digroups which comprise each of the

two mission bit streams and the nature of the failure cause, i.e., if the failure is a result of a known fault elsewhere or unexplained up to this point. Both of these status information needs are simple binary values and can be maintained with minimum memory requirements.

The general concept is to generate, upon detection of a stream status change, a series of messages associated with each bit stream that are routed to the stream's far end along the exact route of the stream. These messages must identify the stream and provide the failed/not-failed, known/unknown information outlined above. These messages are received by each TCU along the route and forwarded through the appropriate branches. As the messages propagate through the network, each TCU adds the information to its own stream status.

With each of these failed/not failed messages, the TCU examines the entire stream associated with the current message. If all of the streams of this group have failed, the TCU can declare a higher level failure. If this higher level failure results in all of the streams failing at even a higher level, the TCU can declare this higher level failure at the same time. This results in the detection of unalarmed higher level faults.

Declarations of stream outages can occur based upon detected failures within the equipment. A frame alarm from both the on-line and standby radio results in the declaration of a link failure independently of any status messages received by a TCU.

Typical operation of this system is as follows. If an unalarmed failure occurs within a level 2 multiplexer, frame alarms are expected from all of the level 1 multiplexers on the transmit side of this multiplexer and carrier group alarms from all of the level 1 multiplexers on the receive side of this multiplexer. If the failure of the level 2

multiplexer is such that the corresponding level 2 multiplexer generates frame alarm on both the on line and standby multiplexer, a mission bit stream failure has clearly occurred at this alarming point.

The TCU responsible for the alarming level 2 multiplexer generates a mission bit stream failure message directed downstream identifying the failed stream and marking the fault as known. The TCU also generates up to 8 digroup failure messages to the TCUs which control the level 1 multiplexers associated with the digroups which pass through this level 2 multiplexer. These messages identify the digroup and mark the failure as known.

Upon receipt of the mission bit stream failure message from the alarming level 2 multiplexer, similar digroup failure messages are generated by the TCU responsible for the non-alarming level 2 multiplexer. These messages are directed upstream to the far end TCUs which control the level 1 multiplexers. To this point, a total of 17 messages have been generated by this failure.

If the failure mechanism is modified such that the mission bit stream remains failed but good sync is transmitted by the faulty level 2 multiplexer, no frame alarm will occur in the far end level 2 multiplexer. After the loss of frame time has elapsed for the level 1 multiplexers that are part of this mission bit stream, the controlling TCUs will generate digroup failure messages marking the failure as unknown. These messages are directed upstream to the digroups' far ends and eventually pass through the receive side of the failing mission bit stream. The TCU which controls this multiplexer will note that all of the digroups that are part of this stream are reporting failures and will declare a mission bit stream failure.

When the mission bit stream failure is finally declared, the identical set of messages are generated as outlined for the alarmed failure, yielding a maximum total of 25 messages generated for this fialure. Messages of mission bit stream failure sent to the digroup ends are necessary to mark the failure as known so that no attempt is made to begin fault restoral for individual digroup failures.

Stream status reporting is required for both failures and restorals. In the case outlined above, restoral of the failing level 2 multiplexer requires that the TCU that effects the restoration generate a mission bit stream restoral message. Mission bit stream restoral also produces digroup restoral messages which change the status of the known/unknown data base.

A method for routing status reporting messages within the network is presented in Appendix B. From this, a general format for the construction of status reporting messages can be easily derived. One simple format for this is shown in Figure A-2. The salient portions of this message include a byte which identifies the message type, a byte which identifies the stream, and a byte which contains the status. Details of the bytes in the status information field are illustrated. It is important to point out that the detail of these bytes is not important at this time. What is important is the idea that the information to be conveyed can be represented in some reasonable way within the byte boundaries.

To provide some feeling for the operation of the status reporting messages, their use within the network will be illustrated. For the purposes of the following examples,

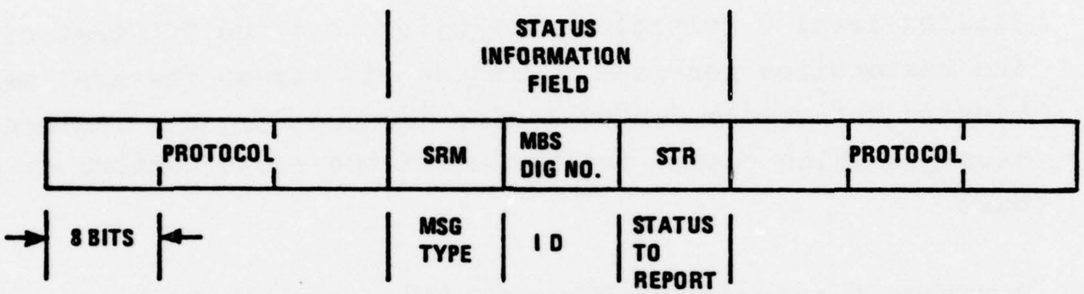


FIGURE A-2
STATUS REPORTING MESSAGE

two models will be used. First is a segment of the DEB network which contains a digroup from Coltano to Donnersberg. This is illustrated in Figure A-3 and is identical to the segment used in Appendix B. Second is an abstract model shown in Figure A-4 that will be used to derive analytical results for system performance. The goal of the analytical model is to provide a very difficult segment that provides pessimistic results.

Some assumptions concerning the operation of the TSC system hardware and software are required before analysis can be attempted. On the basis of the hardware and software presented in this report, the following performance capability is assumed. First, when a status reporting message is received, there is an average delay of 1.5 msec before the station processor acknowledges the interrupt from the telemetry channel hardware. Second, 1.5 msec is required to process each received message. Third, 2 msec is required to reroute the message and perform the fault correlation function. Fourth, an average of 2 msec is required to access the telemetry channel for messages that are ready to transmit. Propagation delays associated with RF path length have not been included.

As will be shown thoughout this appendix within the examples, the status reporting message scheme creates short periods with large amounts of message activity. This implies that there will be contention for TSC resources by these messages. The result of this contention is to delay messages which increase their propagation time. Over the DEB network segment, contention will be assumed to increase the propagation

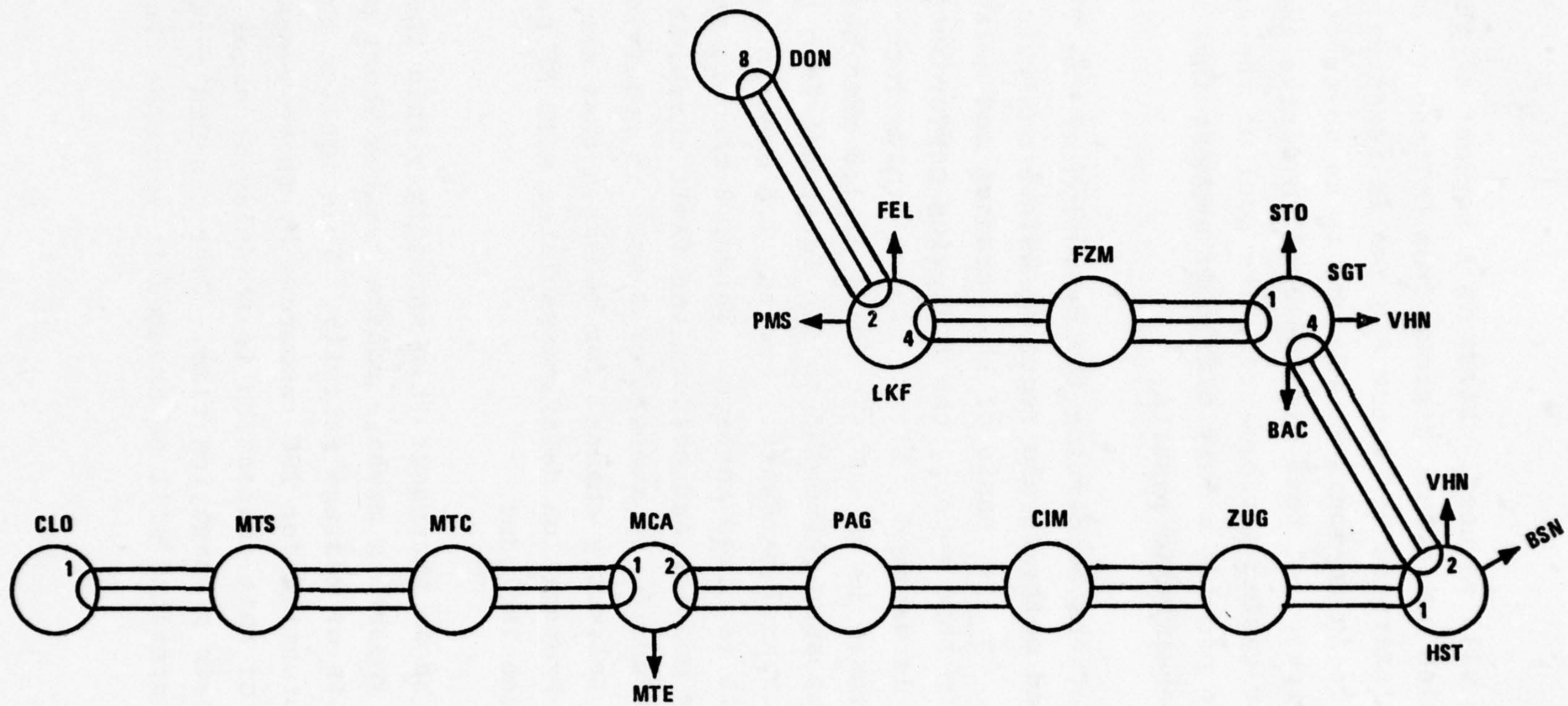


FIGURE A-3
NETWORK SEGMENT - DONNERSBERG TO COLTANO

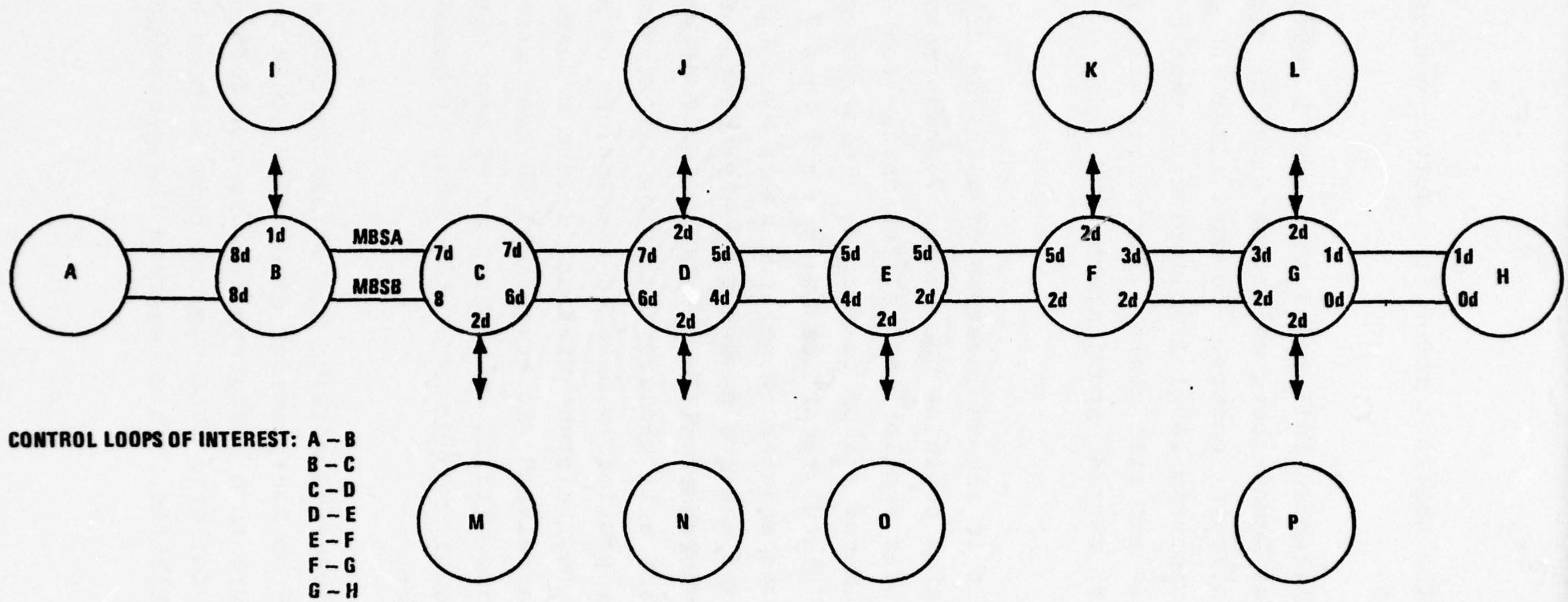


FIGURE A-4

ABSTRACT MODEL

time by 50%. In the abstract model, the actual contention will be determined.

If we assume an unalarmed failure in the level 2 multiplexer common equipment at Donnersberg which contains the digroup at Coltano, the following occurs. 50 msec after the equipment failure, the 7 downstream level 1 multiplexers that compose this mission bit stream will alarm. The alarms will include loss of frame alarm, carrier group alarm, and frame error alarm.

These alarms result in the declaration of a failed digroup by each station which contains one of the 7 downstream level 1 multiplexers. This results in a status change for each of these digroups from not failed to failed. This change in status requires a status reporting message and thus 7 status reporting messages are generated which eventually arrive at Langerkopf and are passed to Donnersberg. Both Langerkopf and Donnersberg will correlate the 7 digroup failure messages and will declare a mission bit stream failure. From the previously defined assumptions of processing time, the most distant digroup failure message from Coltano will be correlated at Langerkopf 40 msec after detection at Coltano with no contention or 60 msec with contention. The same correlation will occur at Donnersberg about 10 msec later.

At this point, an inferred mission bit stream failure exists in the network. From the previous discussion, this is a higher level failure than a digroup failure and further action on digroup failures that compose this mission bit stream must be inhibited. This requires the generation of

additional information which must now propagate to the stream ends which inhibit action. This may be accomplished in 2 ways. The first way is to generate separate status reporting messages for each affected digroup for a total of 7 status reporting messages in this case. The other way is to combine all of the status information fields into a single message which is broken apart at stations which have changes in routing.

The separate message approach possesses some disadvantages. First, the telemetry channel utilization is increased because of protocol overhead. Second, the processing time and processor utilization are increased since each message must be separately processed and interpreted. On the basis of the analysis assumptions, an extra 10 msec of processing is required at Langerkopf for message processing.

A combined message is shown in Figure A-5 as it might be generated. The approach shown in this figure is a very straightforward combining of status information fields and can be optimized in several ways should it be important to do so. For the purposes of subsequent analysis, this represents a very significant improvement.

The combined message is the more desirable of the two methods for the groups of status reporting messages that are associated with mission bit stream and link status change. The overheads associated with protocols and general message processing are reduced and the software required to handle this combined message is only very slightly different than that required to process messages with only a single status information field. The decrease of processor utilization

PROTOCOL	SIF 1	SIF 2	SIF 3	SIF 4	SIF 5	SIF 6	PROTOCOL

SIF ≡ STATUS INFORMATION FIELD

FIGURE A-5

COMBINED STATUS REPORTING MESSAGE

is more than sufficient justification for the selection of the combined message method.

Processing of these combined messages is very straightforward. Generation of the message consists of scanning the routing table for the associated mission bit stream. For each digroup that exists, the three byte status information field is generated. Of the three bytes in this field, only the middle byte is changed and the loop variable which scans the routing table can be used. When the field is composed, it is concatenated to a string which will be the eventual message. After the last digroup is added to the string, the branch far end address is added to the beginning of the message string along with the protocol control field. This forms the complete message ready for transmission.

Similarily simple processing is possible for a TCU which received a combined message. This is discussed later as part of the fault isolation procedure and will not be expanded here. The result of both of the processing methods is to so insignificantly increase the processing time of the status information field that the same processing time assumptions for single status information fields can be used for the combined messages.

Having now defined the operation of the status reporting messages, an overview of the fault isolation and service restoral algorithm can be presented. In general, the fault isolation and restoral algorithm proceeds in three phases. Upon detection of a fault, status reporting messages are generated which eventually lead to the determination of the highest level failure that could cause this fault through the correlation function. When this highest level failure has been determined and control assumed by the appropriate local control loop,

local data acquisition information is used along with the result of the correlation function to build a complete failure syndrome. This failure syndrome specifies a list of appropriate switching actions which are begun. After the switching action list has been exhausted or service has been restored, reports of the action taken and the results of that action are generated for use by the TCU personnel. When service is restored by either the automatic switching of equipment or by manual means (operator directed switching or repair), a series of status reporting messages are generated which lead to the determination of service restoral.

The period of time for the three phases will be refered to as an episode. Because of the nature of the network and equipment, hardware, and software, there is no prescribed ordering of these three phases with respect to their beginning and end. Under certain failure conditions, it is very possible that all three phases could be active simultaneously. The three phases will be referred to as correlation, switching, and restoral in that order.

There are a total of 5 major failure modes that can be detected by the algorithm. These are: alarmed link failure; unalarmed link failure; alarmed mission bit stream failure; unalarmed mission bit stream failure; and alarmed digroup failure in order of decreasing precedence. An unalarmed digroup failure can exist. However, almost by definition, this failure will not be detected by the TSC system without some outside additional information. This will be discussed further under status message processing.

Total message traffic is a function of the hierarchial level of the failure. Assuming no unused facilities, a digroup failure results in a single status reporting messages. A mission bit stream failure will have eight status reporting messages associated with the digroup failure, a mission bit stream failure message between the ends of the failing mission bit stream, and a combined status-reporting message directed to the digroup ends from both sides of the failure. A link failure will produce a message between the two TCUs which contains the link and two mission bit stream status reporting messages directed to the digroup ends from both sides of the failure. This traffic occurs with both the failure and restoral of the affected stream

There are numerous important parameters which must be determined. The amount of time that is required for these messages to propagate throughout the system is certainly important. Equally important is the loading that these messages place upon the system. Both the telemetry channel utilization and processor utilization are vital considerations.

To estimate these utilization factors, a simulation, written in GPSS, was conducted for the analytical model of Figure A-4. The goal of the simulation was to derive an initial set of propagation times, channel and processor utilizations, and some indication of message queueing on a station by station basis.

The assumptions used to drive this simulation model are identical to those previously outlined in this appendix. The mission bit stream status reporting messages and the link status reporting messages were not included. The combined messages were used with either 8 or 16 status information fields, depending upon the failure.

A simulation was run for each of the 4 failure modes listed in Table A-1. The simulation was confined to the message periods resulting from the detection of the failure. The times shown in the table are referenced from this detection of failure point.

TABLE A-1 STATUS REPORTING MESSAGE SIMULATION RESULTS

	MESSAGE PROPAGATION TIME			MESSAGE PEAK LOADING			
FAILURE TYPE	DIGROUP OUTAGE H-B	DIGROUP OUTAGE H-A	HIGH LEVEL FAILURE RPT BACK	MAXIMUM NUMBER WAITING	AVE.DELAY AT MAX STATION	TELEMETRY CHANNEL PEAK USE	STATION PROC. PEAK USE
Unalarmed Link	86 ms	94 ms	258 ms	5(STND)	19 ms	33 ms	90 ms
Alarmed Link	119 ms	126 ms	173 ms	9(STNB)	27 ms	33 ms	90 ms
Unalarmed MBS	57 ms	63 ms	166 ms	3(STNF)	11 ms	17 ms	46 ms
Alarmed MBS	73 ms	79 ms	113 ms	5(STNC)	17 ms	17 ms	46 ms

Several conclusions can be drawn from these figures. First the load placed upon the telemetry channel through the status reporting is minimal. The simulation model is very pessimistic in that all traffic is generated in all cases. Digroup failure messages are not delayed and suppressed in any cases as would potentially occur in the two alarmed failure conditions. Secondly, the simulation model delays and queues messages at the station level and does not allow multiple messages to wait at the processor. The complete message pro-

cessing activity of a status message is performed prior to beginning any activity on any others. This increases the delay time of the message substantially, especially at busy stations. With the processing scheme for these status reporting messages, the station processor is the eventual limiting factor in the overall throughput of the system. This suggests some structuring to optimize the TCU hardware and software to increase the message throughput rate.

Any repair procedure within the DEB network can be viewed as proceeding in a manner as illustrated in Figure A-6. In general, all of the various levels of this restoral tree must be passed through, even though it may not at first appear as though this occurs.

It is important to note that the quality of the information generated by the data acquisition hardware is variable. Certain alarms that are part of the DRAMA equipment require less additional information. Therefore, there is a quantifiable quality factor associated with each alarm which is the number of failure levels that this alarm allows to be tranversed in the failure tree.

Further, there is a similar quality factor associated with each of the deductions that the TCU processor makes. The processor deductions are based upon both the state of remote equipment (or invisible data streams) and local data acquisition information.

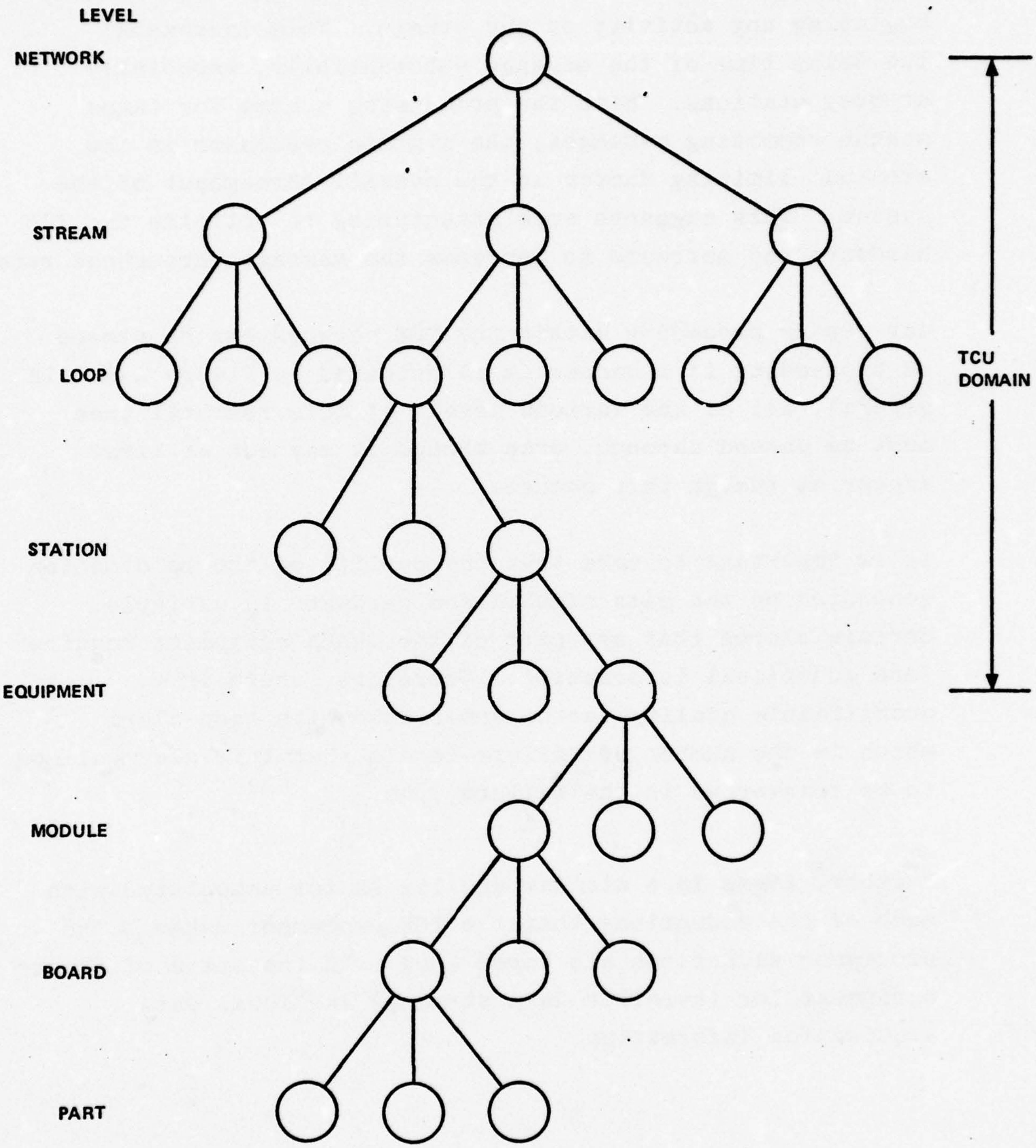


FIGURE A-6
RESTORAL TREE

When all of the possible deductions have been made and data acquisition information considered, it is possible that the lowest TCU level in the restoral tree has not been reached. From this point, the last available action is an orderly switching of equipment, aimed at traversing additional levels of the restoral tree.

It is important to note that, in many failure cases, redundant information exists which allow identical levels of the restoral tree to be traversed. For example, given a high quality alarm such as a primary power failure alarm, the restoral tree can be entirely traversed over the TCU domain. The stream status messages associated with this failure and other potential local alrms will provide no additional information. No simply way seems available to remove this redundant information and it is unlikely that the removal of this redundant information would significantly impact the performance of the TCU system.

It is equally as important to note that additional action is required at each of the levels of the restoral tree. This is particularily true with respect to the message traffic of the status reporting messages. These messages serve two functions within the network. They were first devised to detect unalarmed failures, however, they are equally as important in their task of suppressing actions in other areas of the network that are affected by a failure.

Therefore, each level of the restoral tree must be traversed and the action associated with that level taken, regardless of the quality of the information. If the information is of high quality such that a number of levels may be traversed immediately, this is allowed. In general, additional lower quality information will also exist. If this additional

information does not cause a higher level failure path to appear, work performed in reaching the current level will not be repeated.

From this perspective, the fault isolation and restoral algorithm was developed. The goal of the algorithm is to restore service automatically. If this is not possible, then the second goal is to define the failure to a single piece of equipment. If this is not possible, then the goal becomes to determine the failure over as narrow a range of equipment as possible.

In keeping with the overall organization of the functions within the TCU, the various functions required to perform the fault isolation and restoral procedure are partitioned into the tasks of the system. As information is passed between the various tasks, it is passed in clearly defined, standardized formats to allow standardized software within tasks. In some instances, it is necessary that one task have access to information generated and maintained by another task. In general, this information will not be altered by these accesses.

The restoral tree fully qualifies the failure. Each level of the tree represents a qualification or an attribute of the failure. If a fully-qualified failure is examined, it can be seen that the order of the attributes is not important. The levels of the tree may be interchanged very freely. However, when the overall problem is observed, some organizations are clearly better than others. The most

obvious organization is to proceed from the general to the specific. Outside of a network failure, which is totally inclusive, the most general failure is a stream failure. Therefore, this has been taken as the orientation of the algorithm.

Throughout the algorithm, there is an imperative that no outages be introduced by TCU operations. Any actions taken must leave the network in no worse condition than had no action been taken.

A simplified view of the functional relationships within the fault isolation and restoral action is shown in Figure A-7. The primary entry (and only entry) to this procedure is through a status change report from the three sources shown. After the appropriate message routing has been done, the status change is examined by the correlation function to determine if action is required at this site. If action is required, an action list which is related to the goodness of the alarm and status reporting message information is selected and the required action is performed as part of the equipment switching and result reporting function. The other portion of this function creates information which will assist the operations personnel in their maintenance of the network.

Entries into the status message processing are in the form of entries that indicate current stream states such as failed/not failed and explained/not explained. These are perhaps best represented by the status reporting messages. A second class of information which must be maintained by the local loop is the state of standby equipment. This is particularly true of unalarmed failed standby equipment. The primary goal of operator messages is to convey this type of information.

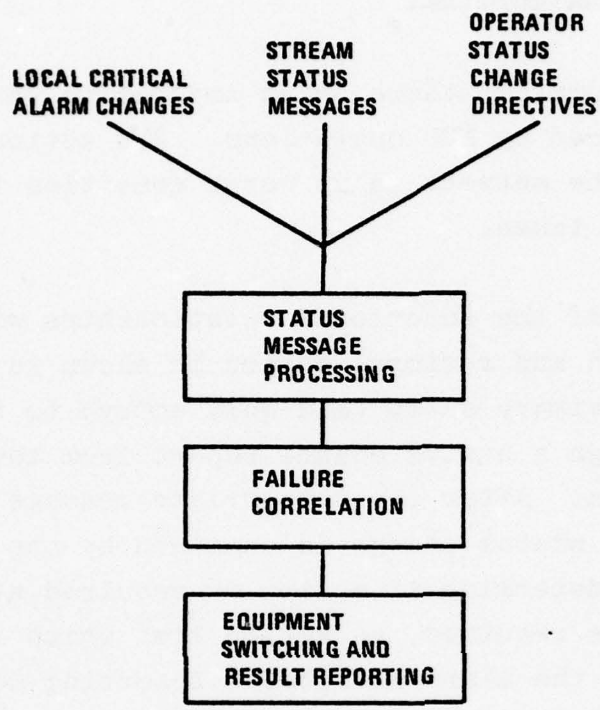


FIGURE A-7

FAULT ISOLATION AND RESTORAL OVERVIEW

One subtle and possible confusing entry into the procedure is the critical alarm change. This is derived as part of the data acquisition task and is cast into a form such that it appears to be a normal stream status change to allow consistant processing. Part of the function of the data acquisition task is to generate this information after viewing the actual alarm change information from the equipment. This includes information that evaluates the goodness of the alarms with respect to the restoral tree.

Failure correlation is the function which attempts to traverse the restoral tree. The organization of this function must be such that it can arrive at reasonable decisions with the weakest information in the simplest way. This is slightly inefficient with strong information since strong information is also requried to pass along the weak path.

Equipment swtiching actions are a result of the level within the restoral tree that the alarm and status information take the failure. Strong efforts have been made to avoid any unnecessary equipment switching actions. Further efforts are made to avoid any potentially degrading actions within the body of the function.

A detailed flow chart of the status message processing function is shown in Figure A-8. One primary action of this function is to compose a list of state changes which will be passed to the failure correlation function for subsequent processing. The second function is to decompose combined status reporting messages and re-route them.

Many of the status reporting messages, especially those which are for streams that do not terminate in the local loop will generate more than one state change that must be processed.

A status reporting message for a digroup which is through-grouped generates one state change associated with the branch on which it was received and a second state change associated with the branch on which it is rerouted.

The data base required for this function is primarily the rerouting tables and the state change lists that it must maintain. Boundaries on the size of the rerouting table have been previously established. Some estimate of the size of the state change lists can be derived as follows. Failures are usually confined to one branch. The probability of two branches failing simultaneously is very small. The worst case failure involves a total of 19 state changes (1 link, 2 mission bit streams, and 16 digroups). If all digropus are through-grouped, the total number of state changes will be 35 for this worst case failure.

Failure correlation is entered from the status message processing function from the "perform fault isolation and restoral." A flow chart for this function is shown in Figure A-9. Flow through this function is oriented to the streams of the system with crossover points as higher level failures become apparent through lower level failures. Points on the flow chart which are labeled "exit" indicate that the processing of the state changes ceases at that point. Either insufficient information exists to continue or the responsibility for continued action is at some other location.

The major information maintained by this function is the failed/not failed, explained/unexplained state of the streams. These are simple binary values which will conveniently fit into single bytes, which will be referred to as bit strips. Use of the bit strips is very simple. If the bit strip of a mission bit stream is considered, the failure of the mission bit

stream can be inferred by testing the byte as a single unit to determine if all of the digroups have failed. A continuation of the restoral action at this point is in order if this failure is not caused by a higher level failure or by a failure elsewhere.

Use of the explained/unexplained state requires some further definition. In this usage, the explained/unexplained state is used as an inhibition to continued fault isolation at a site. This is required to maintain orderly fault isolation and restoral. Two conditions exist for explanation within the system. First, a higher level failure will "explain" all lower level failures. Second, when equipment switching is being performed by one loop on a particular failure, this loop will "explain" the failure until service is restored or until the switching actions have been exhausted.

This function requires access to the routing tables since messages can be can be generated by this function. Also, summary information which must be maintained by the data acquisition talk is required to build a complete syndrom which will be passed to the equipment switching function. Information contained in this syndrome is the stream that has failed, the class of teh failure, and the goodness of the failure information.

Contained within the digroup and mission bit stream failure paths are time delays. These time delays are required to allow other higher level failures to manifest themselves within the network. During the time delay, this function is suspended and the system is free to respond to other state changes. Since a link failure is the highest level failure within the network, no time delay is required to begin its operation.

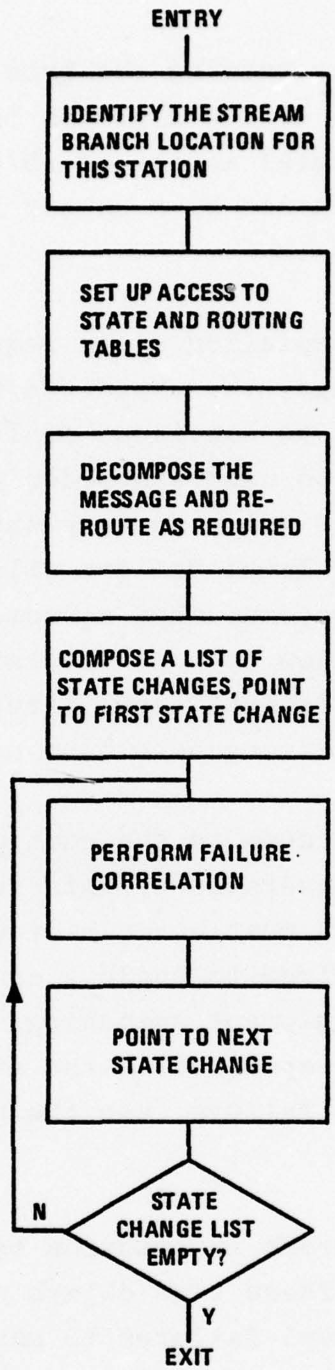


FIGURE A-8
STATUS MESSAGE PROCESSING

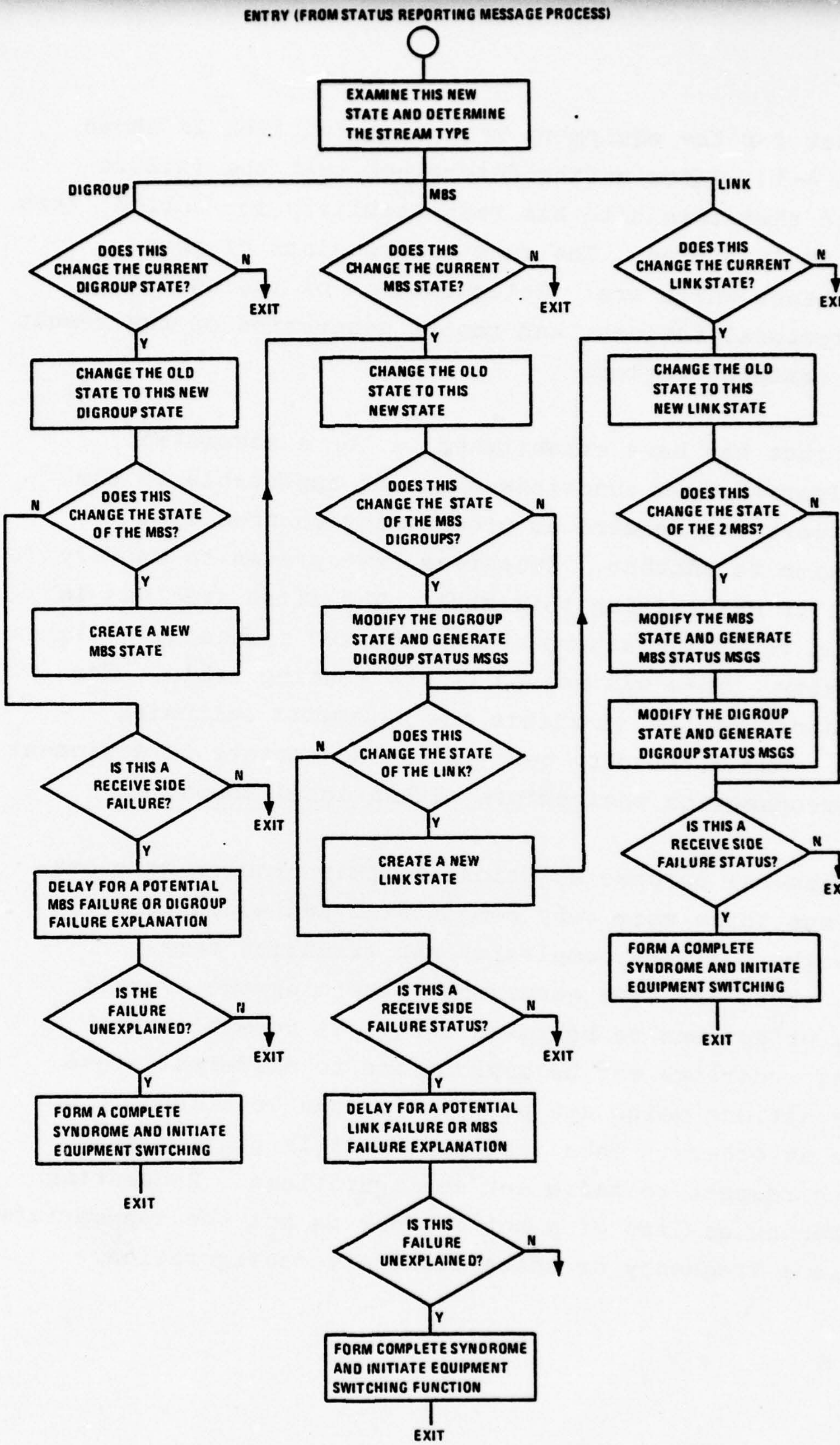


FIGURE A-9
FAULT CORRELATION

A flowchart for the equipment switching function is shown in Figure A-10. After having determined that the failure exists and that this site has responsibility for action, this function is initiated. The function consists of three logical phases which are: determination of the swtiching bounds, restoral actions, and report generation of the result of these restoral actions.

One goal that has been established is for a consistant set of algorithms and functions that are applicable to the whole network and require no programming changes from one location to another. Two areas have proven to be very difficult in maintaining this goal. The first area was in the status reporting scheme which required maintaining digroup connectivity. This was solved by the routing table. The second major problem is within the equipment switching function. The difficulty here is in the variety of equipment and interconnection that exists within local loops.

Several general purpose solutions to this problem were considered but these were very complicated and would tend to increase the software complexity and execution time. Further, there are some questions concerning the overall ordering of actions to be taken and it is possible that different orderings may be appropriate to different areas and some actions which are possible at one location are not possible at others. This last statement is particularly true with respect to radio set configurations. Requesting the switching on-line of a hot standby is not the appropriate action in a frequency or space diversity configuration.

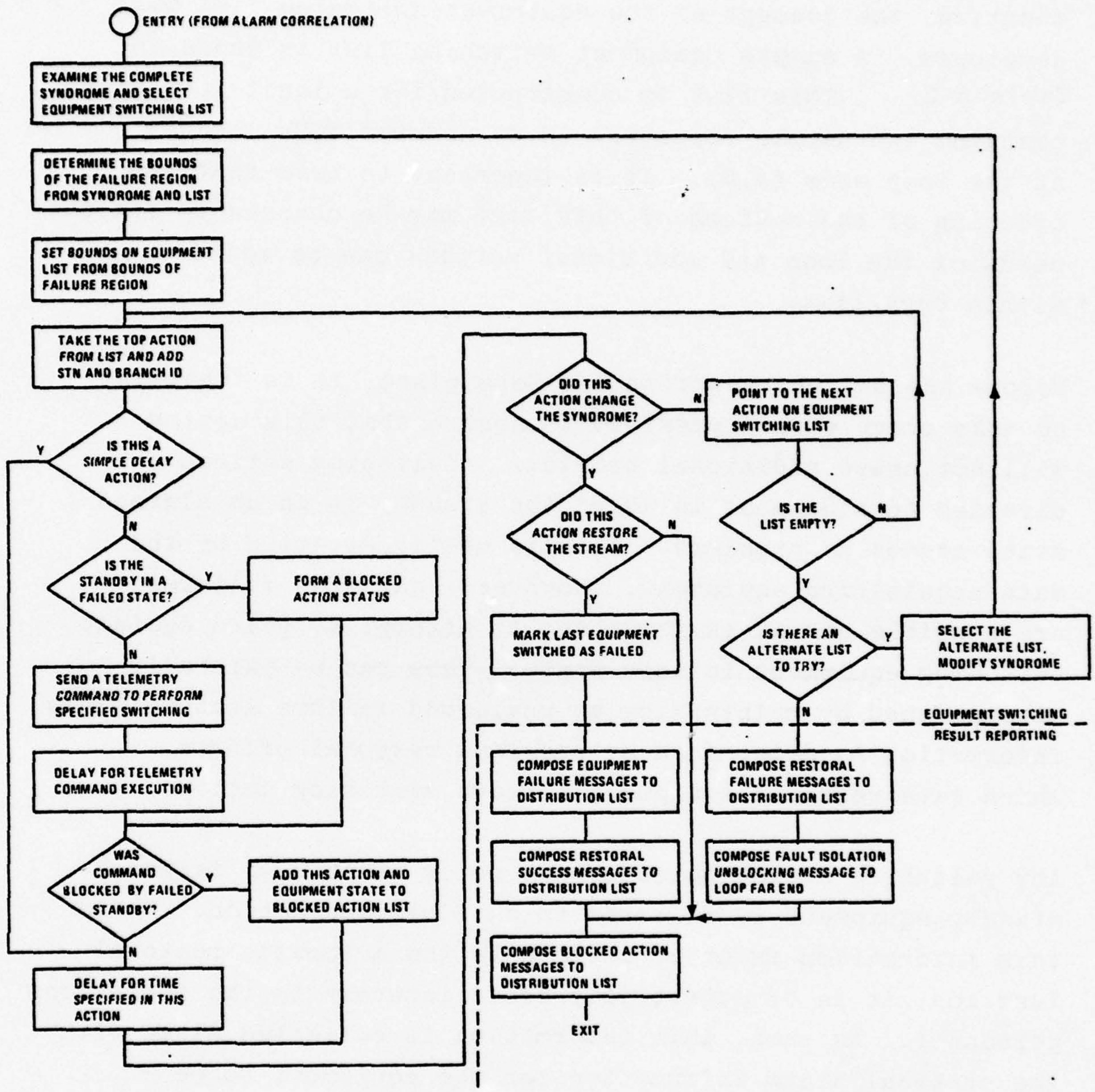


FIGURE A-10
SWITCHING FUNCTION

In order to preserve the generality of the functions and yet to provide the specific information required by this function, the concept of the equipment switching list was developed. A sample equipment switching list is shown in Table A-2. This list is constructed for a local loop which contains two simple repeaters (B,C) and two TCUs located at the loop ends (A,D). It is important to note that the ordering of the actions of this list may be changed to fit the needs of the loop and additional actions can be added within each list.

Before any switching action can take place, it is important to make every effort possible to assure that this action will not cause additional problems. Switching actions directed to equipment in which the standby is in an alarmed state causes no problems. This is easily detected by the data acquisition equipment. However, unalarmed failures are possible and it is important to attempt to guard against switching equipment in this state. This can be partially accomplished by maintaining an unalarmed failure status. This information is determined by previous restoral efforts which have restored equipment through switching actions.

Any switching action which cannot occur because of failed standby equipment is referred to as a blocked action. While this information is of no concern to the automatic restoral function, it is of potentially great interest to the operations personnel. As such, this information is collected along with the critical alarm information for the equipment to be displayed to the operator.

LIST 1	Switch STN A Branch 2 RCVR, delay Switch STN B Branch 1 XMTR, delay Switch STN B Branch 2 RCVR, delay Switch STN C Branch 1 XMTR, delay Switch STN C Branch 2 RCVR, delay Switch STN C Branch 1 XMTR, delay
LISTS 2, 3	Switch STN (near side) Branch 2 RCVR, delay Switch STN (far side) Branch 1 XMTR, delay
LIST 4	Switch STN (near side) Branch 2 RCVR, delay Switch STN (far side) Branch 1 XMTR, delay Switch Nothing, delay long fade period
LIST 5	Switch STN A Branch 2 LVL2 demux (failed MBS),delay Switch STN D Branch 1 LVL2 mux (failed MBS),delay
LIST 6	Switch STN A Branch 2 LVL2 demux (failed MBS),delay Switch STN D Branch 1 LVL2 mux (failed MBS),delay Switch STN A Branch 2 RCVR TDM, delay Switch STN D Branch 1 XMTR TDM, delay Resync STN A Branch 2 KG81 (failed MBS), delay Bypass STN A Branch 2 KG81 (failed MBS), no delay Bypass STN D Branch 1 KG81 (failed MBS), delay
LISTS 7,8	Bypass STN A Branch 2 KG81 (failed MBS), no delay Bypass STN D Branch 1 KG81 (failed MBS), delay
LISTS 9,10	Switch STN A Branch 2 LVL2 demux (failed MBS),delay Switch STN D Branch 1 LVL2 mux (failed digroup, delay

TABLE A-2

EQUIPMENT SWITCHING LIST

After each switching action has occured, there is a time delay requirement. First, equipment must resynchronize and second, status reporting messages must propagate over the network. As was mentioned within the failure correlation function, action within a task is suspended during a time delay so that the system is free to respond to this other information.

In certain segments of the network, some failures are ambiguous especially if the goodness of the failure information is not high. This is especially true in the cases where there are substantial unused resources. To handle these situations, an alternate equipment swtiching list capability is included. The last entry in the primary (or current equipment switching) list contains the information required to activate this list.

Two levels of algorithm result reporting are required. First, as equipment switching lists are exhausted, control must be passed from one local loop to another. Second, there is information which must be given to the operations personnel. Automatic passing of control is comparatively simple, requiring a status reporting message which unblocks action.

There is a substantial amount of information to be passed to the operations personnel. Obviously required is the result of the algorithm within the local loop. If the algorithm restored service, then the last equipment switched is potentially failed and this must certainly be reported.

The distribution of report information is variable and depends upon where the TCU is location within the network. Information is certainly required at the closest manned site

and will likely be of interest to the regional center which is responsible for this TCU. If the control loop spans regional boundaries, the regional center in the adjacent region may require this information. To handle this variability, a distribution list which specifies where reporting information is to be sent is recommended. The anticipated structure of the distribution list would allow complete freedom in distributing the algorithm results.

Performance of the algorithm as a whole was evaluated from the analytical network segment shown in Figure A-4. Along with the assumptions used to derive the status reporting message performance, some additional assumptions are required to evaluate the restoral procedure. These assumptions are as follows. Each switching action requires 25 msec of processing time. Each telemetry command which must be transmitted generates a total of 45 bytes of telemetry channel traffic. Processing associated wtih this traffic is assumed to be contained in the 25 msec of processing time. Delays involved for resynchronization are 50 msec. Digroup propagation delays range from 300 msec maximum to 100 msec minimum. It is also assumed that the last possible switching action restores service.

An alarmed failure in this content is a failure which is alarmed by the failing stream. This is in contrast to an unalarmed failure which is determined by the failure of all streams which compose the failed stream.

Not included in this analysis are any reporting messages other than those required by the automatic fault isolation and restoral algorithm.

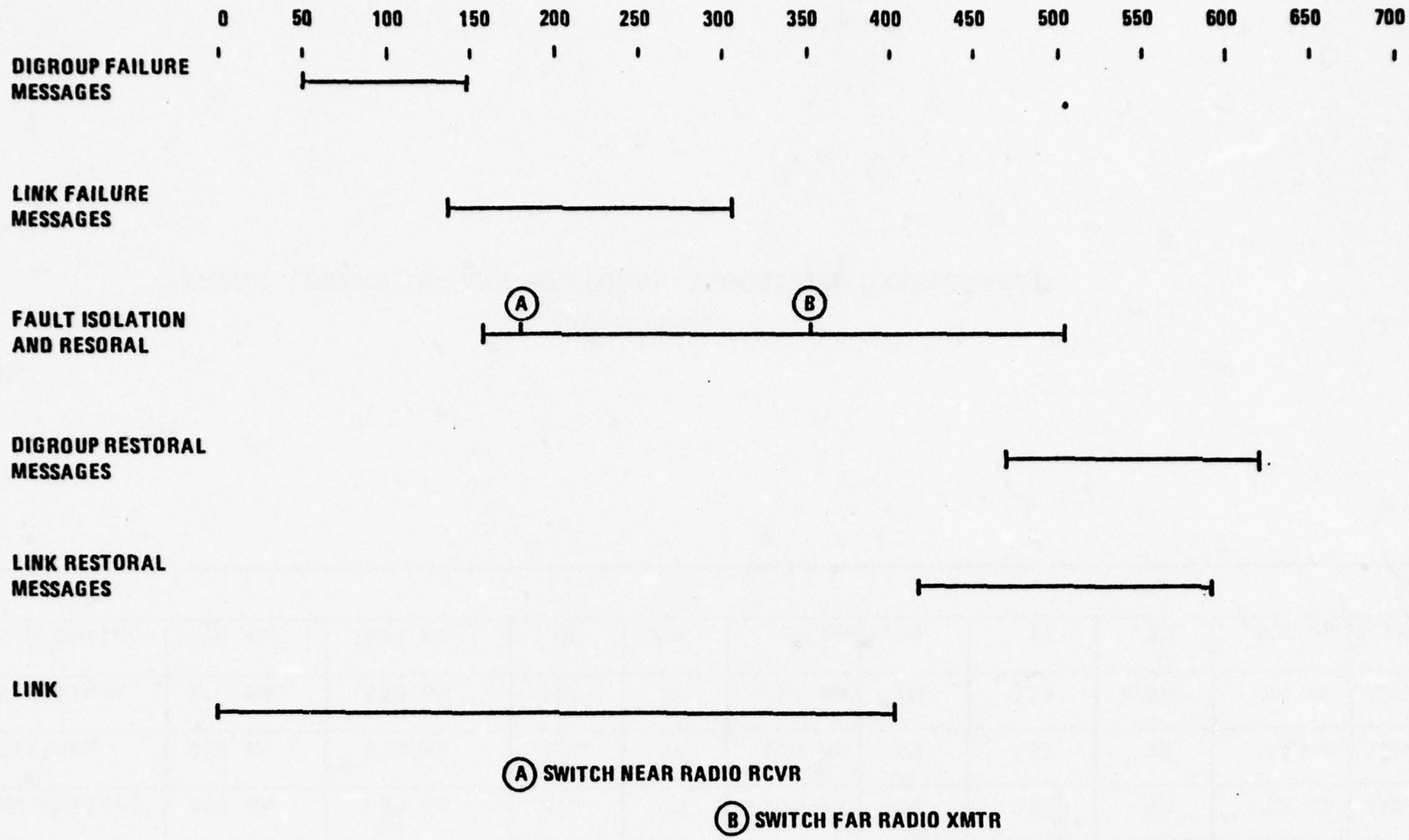
Results of this analysis are summarized in Table A-3 and presented in Figures A-11 through A-16. Percentages of resource utilizations are based upon the episode period as shown in the figures. As can be seen, the overall resource utilizations are comparatively small, especially when viewed in the context of the total episode times. Given the anticipated rate of occurrance of these episodes, the processor resources involved in fault isolation and service restoral are minimal.

As can be seen, the restoral time of the algorithm is determined by a number of variables. The first major variable is the failing stream. A lower level failure requires a longer propogation through the network and must also wait for the declaration of a higher level failure. The second major variable is the length of the switching list associated with the failure. In general, an alarmed failure has a longer equipment switching list than an unalarmed failure. This can be seen from the mission bit stream failures. The last major variable is the number of TCUs which must participate in the restoral procedure. Passing of control from one TCU to another requires additional time.

FAILURE MODE	WORST CASE RESTORAL TIME	FAULT ISOLATION AND RESTORAL EPISODE TIME	PEAK UTILIZATION				AVERAGE UTILIZATION			
			TELEMETRY CHANNEL		PROCESSOR		TELEMETRY CHANNEL		PROCESSOR	
			BYTES	% USE	TIME	% USE	BYTES	% USE	TIME	% USE
Alarmed Link Failure	150 ms	400 ms	372	13%	209 ms	52%	221	8%	112 ms	28%
Unalarmed Link Failure	400 ms	620 ms	372	9%	209 ms	34%	221	5%	112 ms	18%
Alarmed MBS Failure	550 ms	675 ms	234	5%	166 ms	25%	133	3%	70 ms	11%
Unalarmed MBS Failure	410 ms	550 ms	189	5%	121 ms	22%	133	3.5%	64 ms	12%
Alarmed DiGroup Failure	3000 ms	3500 ms	78	.3%	45 ms	1.3%	76	.3%	40 ms	1.2%

TABLE A-3

FAULT ISOLATION AND RESTORAL ALGORITHM PERFORMANCE



OUTAGE PERIOD: 400 ms

EPISODE PERIOD: 620 ms

PEAK LOAD: PROC B 209 ms (34%) LOOP B-C 372 Bytes (9%)

AVERAGE LOAD: PROCESSORS 112 ms (18%) TELEMETRY 221 Bytes (5%)

FIGURE A-11

UNALARMED LINK FAILURE - FAR RADIO TDM MUX FAILURE

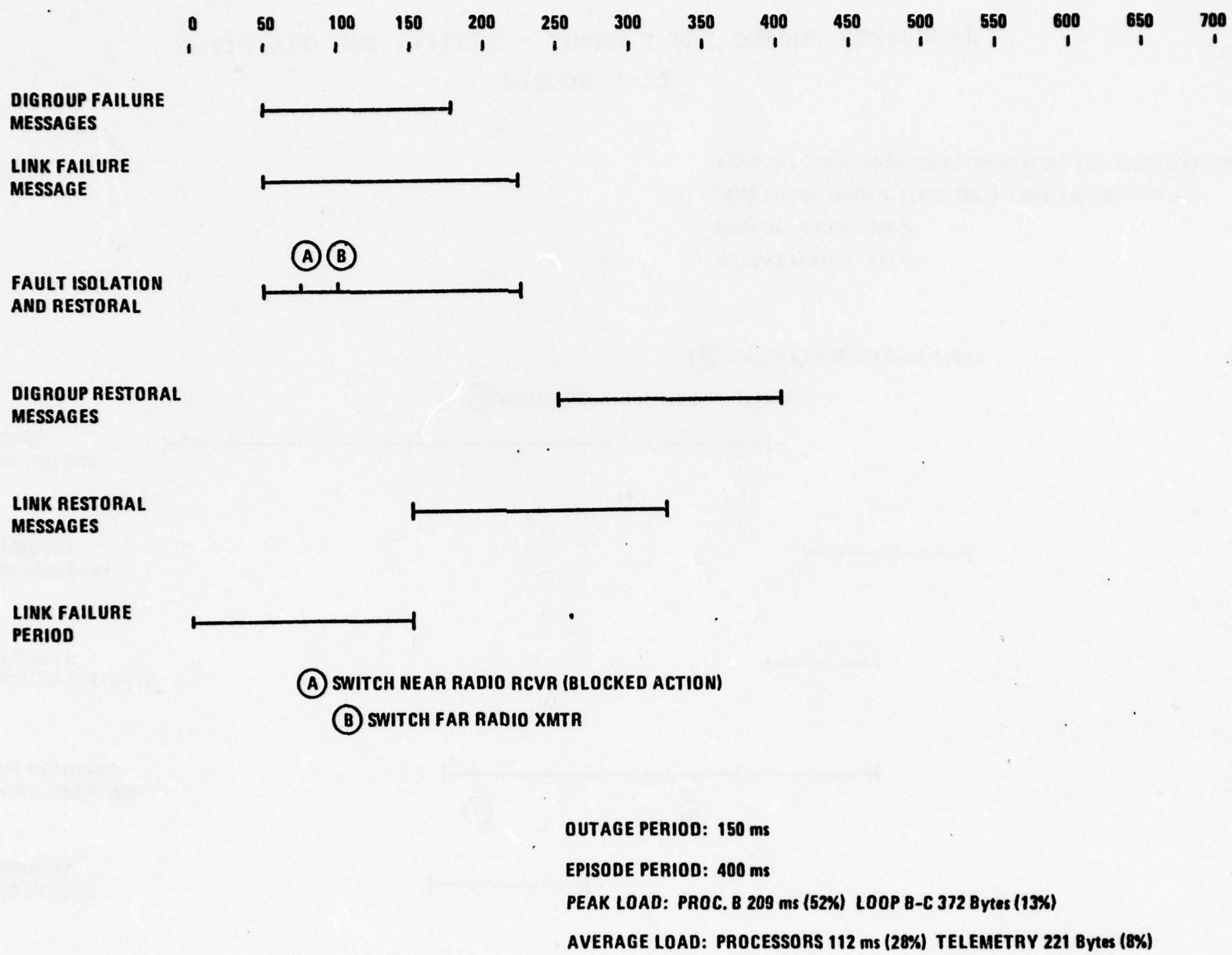


FIGURE A-12

ALARMED LINK FAILURE - FAR RADIO TDM MUX FAILURE

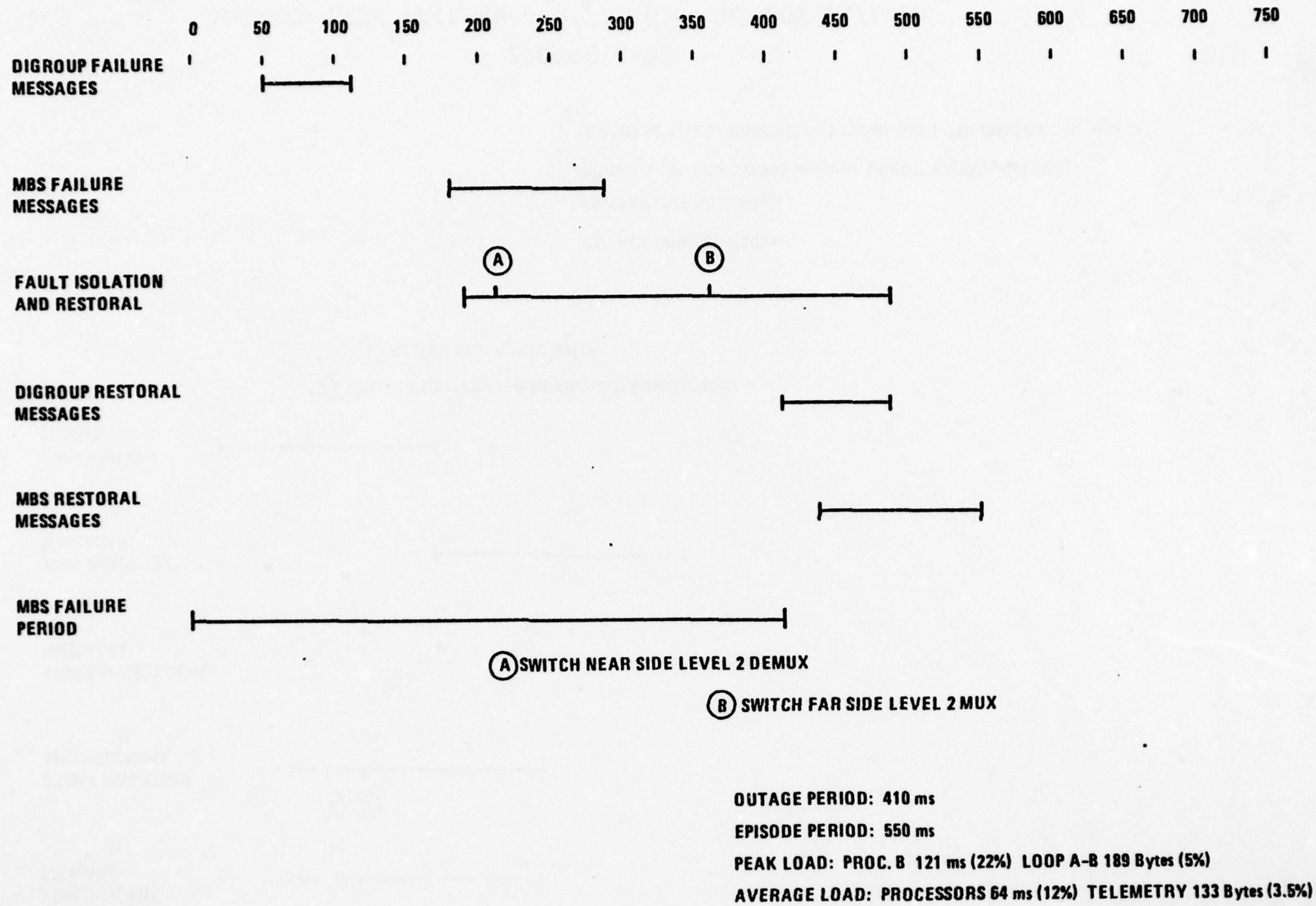
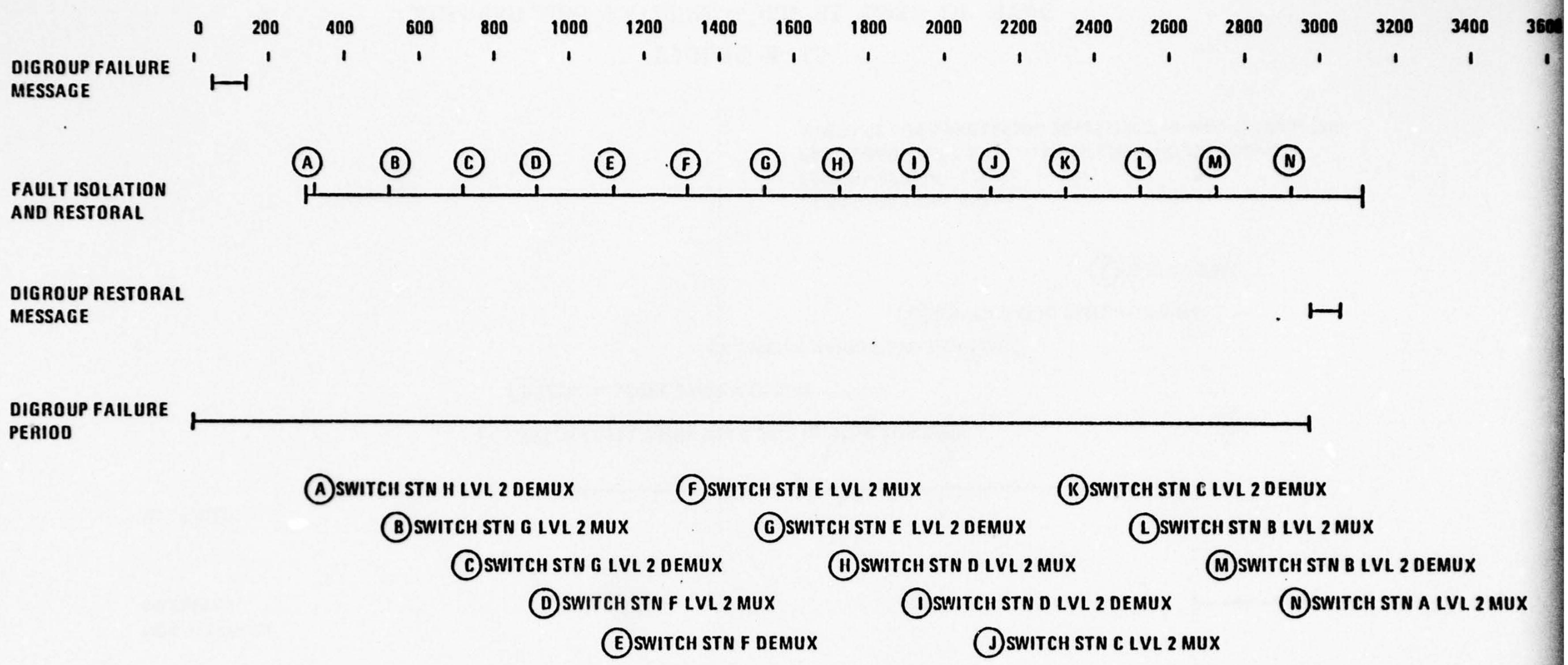


FIGURE A-13

UNALARMED MBS FAILURE - LEVEL 2 MUX COMMON EQUIPMENT



OUTAGE PERIOD: 3000 ms

EPISODE PERIOD: 3500 ms

PEAK LOAD: PROCS. B, C, D, E, F, G 45 ms (1.3%) LOOPS B-C, C-D, D-E, E-F, F-G, G-H 78 Bytes (.3%)

AVERAGE LOAD: PROCESSORS 40 ms (1.2%) TELEMETRY 76 Bytes (.3%)

FIGURE A-14

ALARMED DIGROUP FAILURE - FAR LEVEL 2 MUX PORT FAILURE

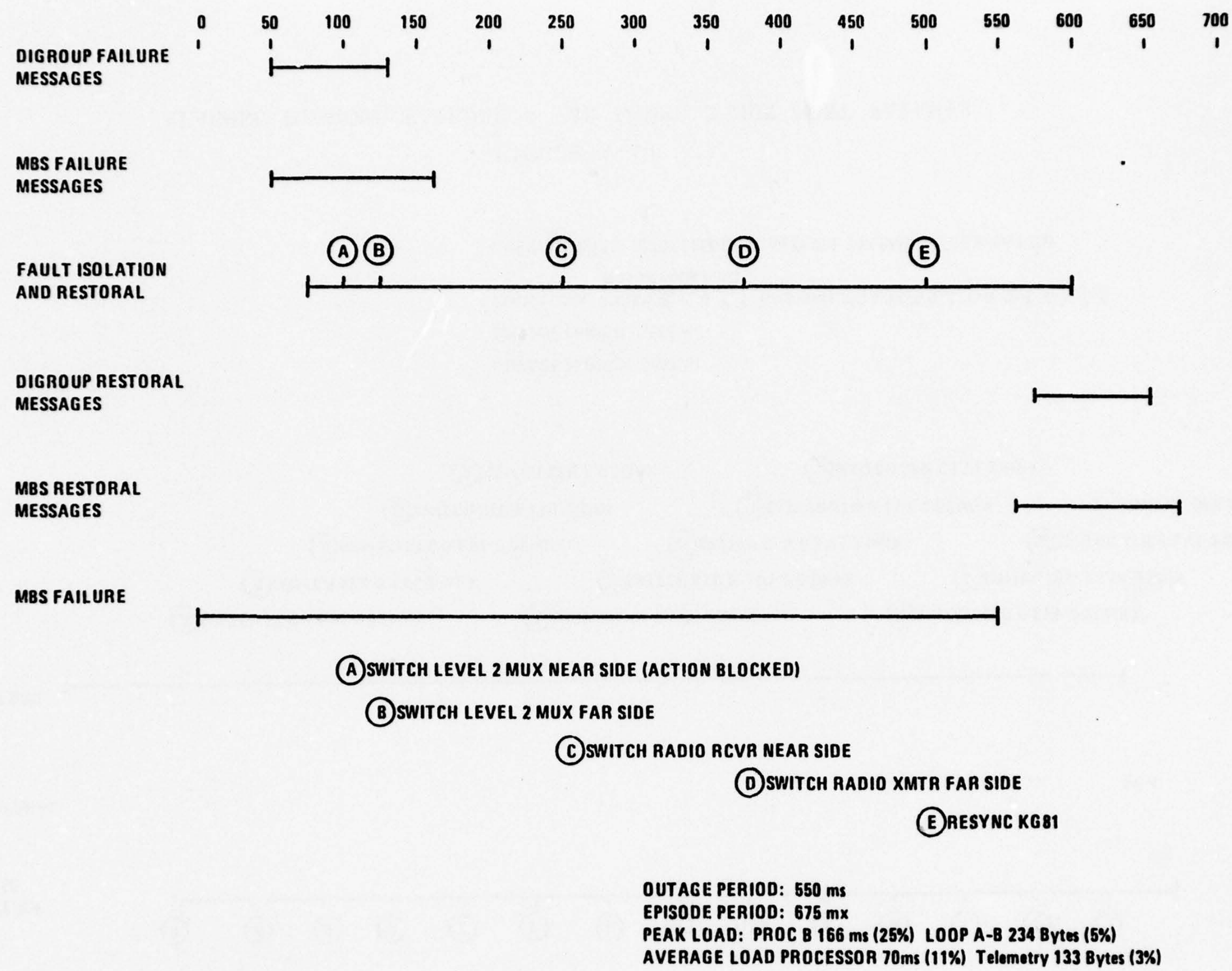


FIGURE A-15

ALARMED MBS FAILURE - KG 81 LOSS OF SYNC

APPENDIX B

A SCHEME FOR DESIGNATING DIGROUP CONNECTIVITY

As discussed throughout this report, failures and restorals are stream oriented. No difficulties are experienced in determining the alarm status of link circuits and mission bit stream circuits. However, the digroup poses some distinct problems. Digroups are not confined to any set pattern of implementation; they cover large numbers of stations and local control loops, and pose hardware and software problems within the TSC system.

There are some potentially serious problems involved in utilizing digroup connectivity, other than those outlined above. Some of the more obvious problems include the following. The data base for digroup connectivity must be maintained by the TSC software. Changes in digroup connectivity involve changes in the connectivity maintained by the TSC software.

The scope of this appendix is to discuss only the methodology of maintaining digroup connectivity and does not include the total utilization of this digroup connectivity or the performance to be gained by using it. These are discussed under the fault isolation and restoral sections.

The purose of maintaining digroup connectivity is to allow messages to be propagated along the exact path of the digroup. The information field contains first some indication of the message type; second, some method of conveying digroup identification; and third, information which is to be conveyed along this path. As is discussed under the fault isolation and restoral algorithm, the information is a single 8-bit byte. The message identification is also a single 8-bit byte.

In general, the suggested method for maintaining digroup connectivity involves minimum impact on the overall TSC system. This method maintains a very compact routing table associated with each station that demultiplexes mission bit streams into digroup streams. Stations which demultiplex but are not TCU stations have their connectivity stored at the loop end TCU's. The purpose of the connectivity tables are to route messages over the exact route the digroup travels and to identify its position within a mission bit stream along the route.

Two methods of designating digroup connectivity were explored. The first method transmits the entire digroup connectivity and routing along with the message. The second method distributes the digroup routing along the path. Several variations of the first method were also explored to increase efficiency.

By including routing information imbedded within the message, no routing decisions are required along the path that the message must travel. This is a conceptually simple scheme which could be implemented reasonably within the network.

It is important to realize that message routing occurs between TCUs and not between stations. In general, as a stream passes through a control loop, there is no modification to its connectivity or routing. In the situations in which there are modifications such as in drop and insert repeaters, the information is still acted upon by the controlling TCU. One of the longer paths found within the network is a digroup between Donnersburg and Coltano. This digroup passes through twelve stations but only through five control loops.

For the purposes of message routing, this digroup will be used. Details of the routing are shown in Figure B-1. To analyze the message traffic, the following will be assumed. Digroups can be associated with level 1 multiplexers and will span from 0 to 10 TSC control loops with an average span of four control loops. Mission bit streams are associated with level 2 multi-

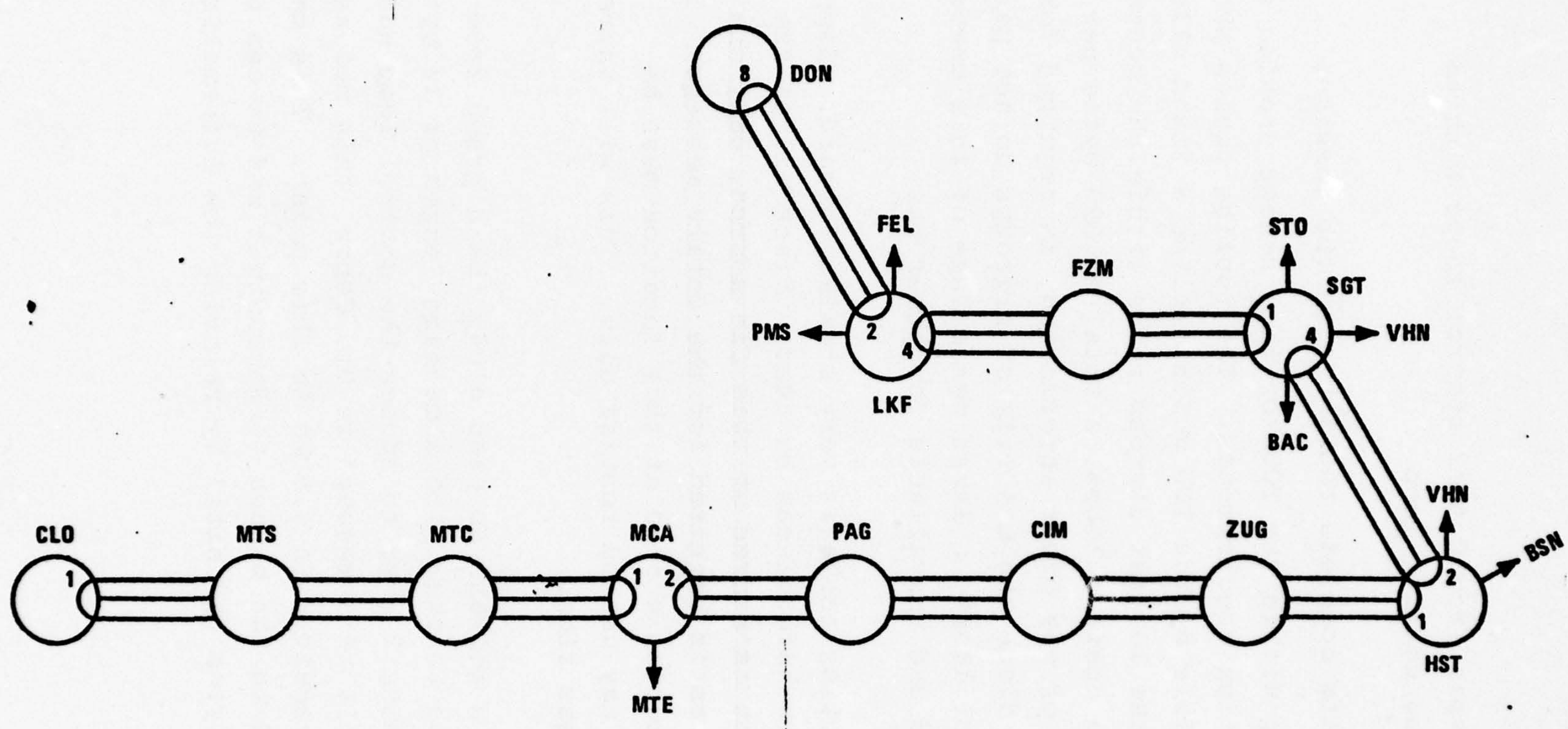


FIGURE B-1

COLTANO TO DONNERSBERG DIGROUP

plexers and will span from 0 to 2 control loops with an average span of one control loop.

One scheme transmits complete routing with the message. Storage of routing within the TCU for this scheme provide a series of conflicting requirements. The routing tables should be rapidly accessible by the TCU which implies a fixed allocation scheme. If the longest digroup route within the network passes through ten control loops, a total of 320 bytes per branch, exclusive of any other information, is required for digroup routing. Since the majority of digroups do not pass through ten control loops, a large percentage of this memory will be unutilized and unavailable for other use.

Access to this routing table is very straightforward. Simple array addressing techniques can be used. Since all of the routing information is stored at this one source, only one routing table access is required for the entire message. The 20 bytes of information stored at this location must be scanned to remove any unused routing data. This will increase the effective access time.

With this scheme, a message to pass along the digroup from Coltano would leave Coltano with a message length of 18 bytes (144 bits). Assuming 2 msec to access the control loop and 2.5 msec to transmit this message to Mt. Cerna, this message will be complete received in 4.5 ms to this point. This message must now be routed to the branch to Hohenstadt and it can be shortened by two bytes (16 bits) by removing the information

pertinant to Mt. Cerna. Allowing 5 msec of internal processing time and 2 msec control loop access time, the message will arrive in Hohenstadt approximately 14 msec after being generated in Coltano. Assuming no contention for the telemetry channel by other message traffic, the message generated in Coltano will arrive and be interpreted in Donnersburg 50 msec after its generation.

A more compact storage mechanism would be to store the routing information in the form of a singly linked list. The overhead associated with this method will be 3 bytes per active digroup. No storage is used for digroups which are not implemented. For a branch which has 16 active digroups which pass through an average of 5 control loops, 208 bytes would be required. No unused storage would exist within this area. Access time to the routing information with the scheme could take as long as 16 linking operations as opposed to the single indexing operation of the fixed allocation scheme.

A sample of this scheme is shown in Figure B-2. Routing table access proceeds as follows. The first location of this list (top) is pointed to. The digroup number for which routing is desired is compared to the stored number. If they match, all of the connectivity stored from this point to the end of list delimiter is the connectivity for this digroup. If there is no match, the next digroup element is pointed to on the basis of the pointer stored with this digroup.

The access time of this method can be increased substantially by the use of an auxiliary table which is accessed through standard array addressing techniques. The values stored in each element of the array are the starting addresses of the routing. Very little additional memory is involved. The only increase is 2 bytes for each unused digroup.

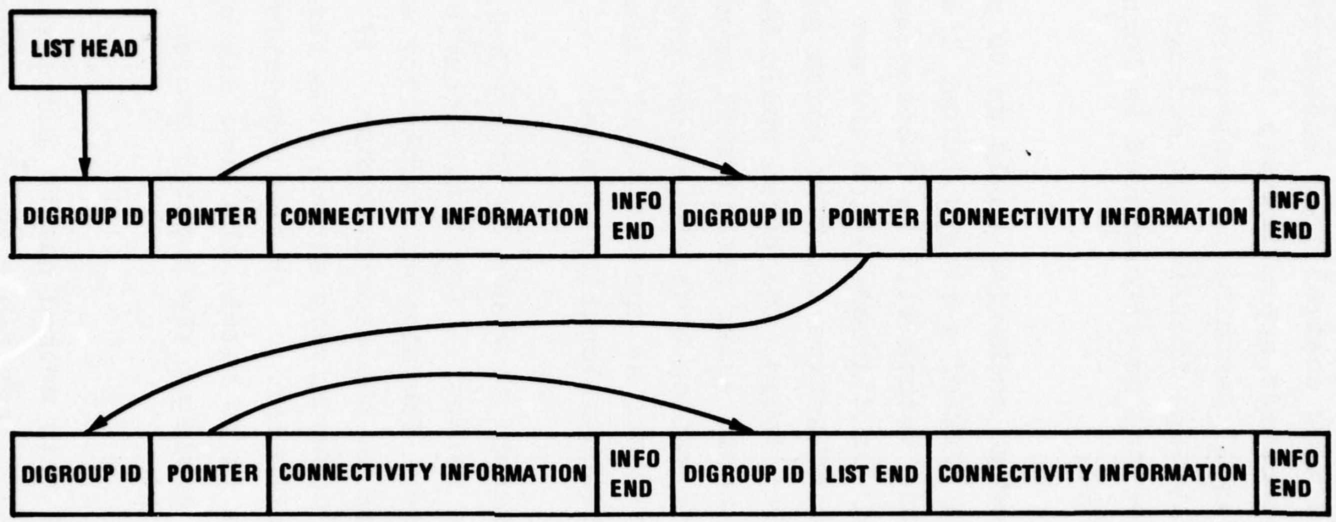


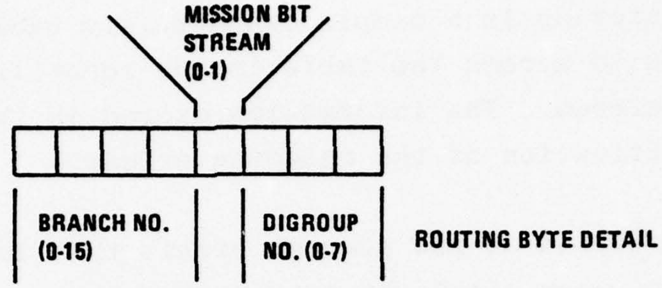
FIGURE B-2

LINKED LIST

When viewing the routing problem over the network, it should be noted that routing is modified only in stations that demultiplex the streams and that within a demultiplexing station, all that is required to specify the routing within that station is a simple connectivity table. The information needed to access the table is the identification of the incoming data stream. The information stored in the table is the identification of the outgoing stream.

Construction of and storage within this local routing table becomes very simple by noting that the station does not need to know any final destinations. All the station is required to know is if it is the final destination or, if it is not the final destination, which branch it must place the message on to route it to its final destination. If there are no stations which will ever exceed 16 branches, the entire set of routing information can be coded into a single 8-bit byte and the largest table that will ever exist (for a station with 16 branches) will be 256 bytes. Details of this table and byte are shown in Figure B-3.

One prime difference between this scheme and the previous scheme is that the overall message length is shorter and constant between the local loops. One piece of information which is not specifically delineated within the message and analysis is that the receiving station knows the branch number on which the message is received. This information is available to the telemetry processor when the message is read from the telemetry channel interface and it is assumed that this is included as part of the message block created by the input routine.



LANGERKOPF CONNECTIVITY TABLE

DIGROUP / BRANCH	MISSION BIT STREAM A 0	1	2	3	4	5	6	7	MISSION BIT STREAM B 0	1	2	3	4	5	6	7	
1 (PMS)	(1) 1A0	(1) 1A1	(1) 1A2	(1,2) 3AD	2B7	2B6	2B5	(3) 1A7	4B0	4B1	4B2	(1) 1B3	(1) 1B4	(1) 1B5	(1) 1B6	(1,2) 4A1	CONNECTION
2 (DON)	(2) 1A3	4B3	4B4	4B5	(3) 2A4	4A5	4A6	4A7	(1) 2B0	(1) 2B1	(1) 2B2	(1) 2B3	(3) 2B4	1A6	1A5	1A4	
3 (FEL)	(3) 3A0	(3) 3A1	(3) 3A2	(3) 3A3	(1,2) 3A4	(1) 3A5	(1) 3A6	(1) 3A7	(3,4) 3B0	(3,4) 3B1	(3,4) 3B2	(3,4) 3B3	(3,4) 3B4	(3,4) 3B5	(3,4) 3B6	(3,4) 3B7	
4 (SGT)	(1) 4AD	(1,2) 1B8	(1) 4A2	(1) 4A3	(3) 4A4	2A5	2A6	2A7	1B0	1B1	1B2	2A1	2A2	2A3	(1) 4B6	(1) 4B7	

NOTES:
1. LEVEL 1 MULTIPLEXER EXISTS AT THIS STATION FOR THIS DIGROUP
2. THIS DIGROUP IS SPLIT
3. THIS DIGROUP IS NOT USED
4. MISSION BIT STREAM B FROM LKF TO FEL IS NOT USED

FIGURE B-3
LOCAL ROUTING TABLE

Access time for this method is very rapid. Simple array addressing can be used. Since the array information is a single byte, large movements of data are not required and no scaling is required during the address calculation.

Comparison between the two methods is direct. The imbedded routing method requires the greatest amount of memory for routing storage, has the highest telemetry channel utilization since the messages are longer, and is not as flexible in implementing changes in network connectivity. The distributed routing method requires only very modest amounts of memory, increases telemetry channel utilization very slightly, and can be organized to allow for changes in network connectivity from either site supplied information or remote telemetry command.

There are several important additional considerations which are necessary for a complete scheme of digroup connectivity. The primary purpose of these messages is to direct information from one stream end to another. In the case of a digroup the individual VF channels are frequently split within the network so that a digroup may have more than one end. Second, some means is required to identify the end of a digroup path. Third, some means of identifying unused digroups within a mission bit stream is required.

While there are numerous ways of providing this information within the system, the most direct method is to provide an additional table which contains this information. Since the desire is to provide a rapid access system, this table will have a minimum size of 1 8-bit byte per entry. Five bits of this byte can be used to designate the number of VF channels that terminate at this station. An entry of zero VF channels would

designate the digroup does not exist. The remaining 3 bits can be utilized to store other information about the digroup.

In the case of split digroups, the routing table must direct the information to the most distant point. The additional table will contain the number of VF channesl which terminate along the route. The end point of all digroups either split or complete is easily contained within the routing table. The simplest convention for this is to terminate the routing upon itself. If the routing address contained in the routing table is equal to the routing address used to access the table, the stream terminates at this point.

The total requirements for maintaining digroup connectivity are relatively small. A 16-byte routing table and a 16-byte additional information table is required for each branch which demultiplexes a mission bit stream. In the situation in which a drop and insert repeater is contained inside a local control loop, this information must be duplicated at the TCU ends but it is not required at the drop and insert repeater. The software required to perform the routing function and to maintain the routing tables is simple and has a very minor impact on the overall size of the TCU software package.

APPENDIX C

SOFTWARE ORGANIZATION

The general software organization is a series of modules which represent a logical partition of the tasks required within the system. The software falls into two general catagories: specific functions to perform the desired TSC mission, and modules which are required to support these mission functions. Some additional general characteristics are that the majority of the functions are task oriented, some of the modules require a re-entrant capability, and the general software is interrupt driven.

Details of the mission functions (telemetry, data acquisition, and fault isolation) are found in their respective discussion areas. The purpose of this appendix is to discuss the design philosophy of the software and to show the operation of and interaction between all of the modules required for an operational system.

Some of the more major ideas of structured programming have been used in the development of the software concepts. Since no coding has been performed in algorithm development, details of coding such as "goto-less" programming have no bearing at this time. The relavent portions of structured programming are: each function or unit of code has a single entry point and a single exit point; each major function is decomposed into smaller functions and these are decomposed until a machine language code is generated; the entry point of each function has a clearly defined set of inputs; each function preserves some relationship such that it must eventually complete its operation.

The first major task in software organization is the partitioning of the software into a reasonable set of software modules. The problems involved in software partitioning are to minimize the total memory required and to maximize the execution time. Another major consideration is to modularize the software such that changes within one section do not impact (or minimally impact) other functions.

Partitioning was accomplished by listing all of the functions required to perform each of the major missions and estimating the relationships of occurances within the system. This was refined several times to create the final partition. Each of the major missions have been partitioned as separate functions based upon the stimuli which trigger their action. A number of support functions were created on the basis of their applicability to two or more mission functions and were partitioned out to avoid duplication. A small number were partitioned into functions because of the anticipated timing relationships within the system. The final partition is shown in Figure C-1.

The system software is interrupt driven. All of the functions are dormant until specific external stimuli occur. At the moment that this interrupt occurs, the function becomes pending, and unless there is a more important function which is currently active, the function becomes active and begins execution. The interrupt activation is a function of the system hardware and is asynchronous to the software. The software control of an interrupt driven system consists of enabling and disabling the interrupt system and determining to what interrupts the system will respond.

Typical practice in interrupt driven systems is to allow the interrupt system to remain enabled as much as possible so that no information is lost by failing to respond and to provide

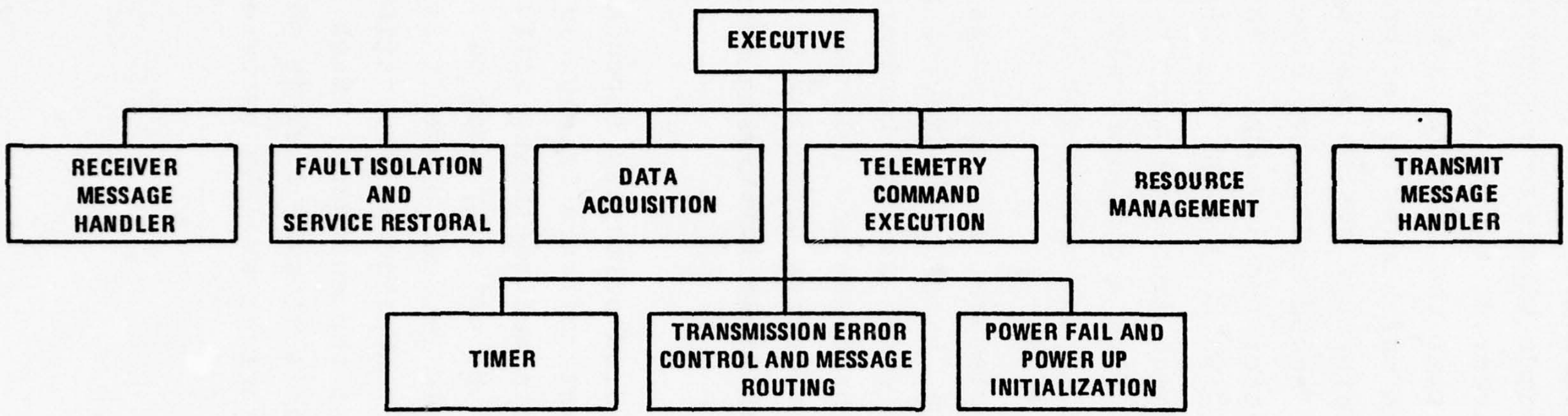


FIGURE C-1

SOFTWARE PARTITIONING

hardware and software buffering in situations where multiple interrupts could exist from a single source which could be lost. The amount of time the interrupt system can remain enabled is related to the amount of time the interupt system must be off. For a given amount of work which must be performed with each interrupt, the time the interrupt system must be off is a function of the processor speed and memory size. Processor speed is obvious. The faster the processor, the faster the interrupt service work can be performed. Memory size is less obvious. Memory size becomes important when the software buffering is considered. Even a comparatively slow processor can perform well with sufficient memory for buffering.

Within the hardware and software, several areas required buffering. Where possible, hardware buffering has been employed. Specific areas which utilize hardware buffering are the data acquisition hardware and telemetry channel interface. The software requires buffering to handle potential overflow conditions and multiple interrupts which are not buffered in hardware.

Software buffering is handled primarily through the property of re-entrancy. Re-entrant software is capable of ceasing activity on one set of data and beginning activity on a new set of data. As soon as the activity ends on the new data, the activity on the old data is resumed where it left off. Re-entrancy is achieved by separating the variable storage area from the main body of the software. Each time the software is initiated, a new variable storage area is obtained. When the activity ends, this storage area is returned so that it can be reused.

The task structuring of the software is such that the operation of a module may be suspended by the module and control given to some other section while the suspended module waits for additional information. When this information becomes available, the task can resume its activity at the point at which it was suspended.

This software organization should provide a high degree of rapid response within the TSC. Little effort is wasted on unproductive polling. Each module returns control as rapidly as possible and allows interrupts as soon as feasible in its processing.

Suspension and activation of tasks occurs primarily through the executive which also provides one of the means of communication between the various tasks. Each request by tasks for processor resources of time and memory are routed through the executive. As memory resources are no longer required by a task, they are returned to the available memory pool through the executive. When there are no active tasks within the system, the executive is used as the active task. The primary job of the executive is to maintain the orderly execution of each of the tasks. As such, it must maintain the state (active, suspended, dormant) of each of the system modules and the conditions required to reactivate suspended tasks, and to activate dormant tasks as required bv the system state and by other tasks. The executive must maintain the variable storage areas of suspended tasks and lists of information to be communication between tasks.

Except for special circumstances such as maintenance and repair activity and diagnostic software execution, executive functions are fixed as are the total module complement. This allows the executive module to be tailored to the system and written to take advantage of this fixed environment. Since the executive has a high degree of utilization within this scheme and respresents

an overhead function as opposed to an application function, time spent in the executive must be minimized. A desirable goal would be to structure the executive such that the majority of its operations can be performed in fewer than 100 instructions. For the majority of the tasks, this should be possible.

Given the potential range of memory requirements both in terms of software buffering and size and numbers of messages that can exist, fixed memory allocations are not appropriate. Allocating fixed buffer areas to handle worst case situations is wasteful of memory if these worst case situations are infrequent. Allocating fixed buffers to handle average cases will prove catastrophic in worst case situations. What is required is dynamic buffering to allocate memory from a memory pool on an as needed basis. Buffer memory requirements vary between the tasks of the system depending on the number of variables that are required by the task and the amount of information which must be processed by the task.

There are a large number of schemes for memory allocation that can be employed. Again, memory management is an overhead function and execution time for this task must be minimized. Also, utilization of memory must be high so that the unused capacity is minimized. Further, it is also desirable that the execution time required for memory management remain relatively constant.

One method of accomplishing these goals is to logically partition the memory pool into small segments and to allow each task to request as many of these segments as it requires. The problem is in selecting the segment size. If the segment size is large, memory utilization is reduced for tasks which do not require large segments. If the segment size is small, the processing time for tasks which require multiple segments is increased due to the increased difficulty in accessing potentially scattered areas of memory.

From approximate estimates of the complexity of the software modules, it appears that the optimum segment size ranges between 64 bytes and 256 bytes. A segment size of 128 bytes should be adequate if the maximum message length of telemetry messages is limited to 1000 bits. This segment size allows for register storage (32 bytes), variable storage (72 bytes) and executive control information (24 bytes) for simple tasks. For complex functions or tasks, multiple segments can be allocated.

Memory management will maintain all of the free areas as a linked list. Each request for memory will allocate the first free memory area to the requesting tasks and establish any linkages that are required by that task to use the freshly allocated area of memory. When the task is finished with the memory segment, the segment is returned to the linked list as free memory.

Timing information and time delays are required in several different areas of the system. Time delays are required in fault isolation and restoral to allow for equipment resynchronization and message propagation. Timing is required within the data acquisition module to test for valid alarm conditions and to delay syndrome building until all alarm conditions for a fault have occurred. Time intervals are needed in transmission error control and telemetry command execution to institute recovery procedures in the event of fault conditions.

Several possibilities exist in the generation of time intervals. The first possibility is to create a hardware timing subsystem. The second is to generate time intervals in software based on a time of day. The third is to create a software interval timer in which a preset time interval is counted down by a software routine. Other possibilities exist in terms of combinations of hardware and software.

A hardware timing subsystem provides the minimum impact on the software system in terms of overall software complexity and software execution time. A hardware timing subsystem provides a simple set and forget operation and implies no software action during periods of timing. A hardware timing subsystem will have an associated

hardware cost for the timing hardware but will reduce the software which will slightly reduce the processor memory requirements and hence, the processor cost. It is unlikely that the decreased processor cost will balance the increase in the hardware timing cost so that the hardware timing subsystem will increase the cost of the TSC hardware.

Any software based timing function must be carefully considered. The flexibility available with a software timing function is very large, however, the amount of computational resources involved can adversely affect the operation of the system. If 20 instructions are required to update each timer (100 usec) and 10 timers are active and required updating, 10% of the available processor resources can be absorbed in the timing function. The variables which will affect the timing function performance are the organization of the timers for software efficiency and the time interval resolution required within the system.

The most efficient software organization is to order the timers sequentially in memory. On the order of 10 instructions are required to update and test for completion for each active time. Much of this efficiency is lost when a timer completes its cycle since it must be removed and the timers shifted so that there are no holes in the sequential group for the next update. Further, it is slightly more difficult to associate the timer with the task since the timer is separated from the task.

Timing can be incorporated into the task's data base and standard list processing techniques can be used. A list oriented updating of software timers would require on the order of 20 instructions to update and test for completion each timer. Removal of the timer from the timing chain is simple and efficient. Further, the number of timers is only limited to the available processing time.

Time interval resolution is a function of the degree of grain that can be tolerated within the system versus the amount of processing time that can be allocated to the timing function. In our review of the system, no need can be shown for a time resolution greater than 5 msec.

Power failure and power up initialization functions are the last of the non-mission or support functions. In both cases, these functions are required to place the TSC hardware and software into known states under extraordinary circumstances. Upon detection of a primary power failure, all TSC control of the DRAMA equipment must be released and the telemetry channel bypassed. Hopefully, a message can be generated indicating the failure.

Power up initialization must set the executive to a known state and collect the station status. Messages indicating the powering up of the TSC hardware at this station must be generated and control of the telemetry channel assumed. At least portions of the initialization routine must be available for operator restarting. This is required after periods of long outage so that stream status can be reacquired and also to allow the operators to force the TSC hardware into a known state.

The remaining software modules are mission oriented. Fault isolation and service restoral and data acquisition represent major missions of the TSC hardware. The receive message handler is a separate task since messages will be received asynchronously with respect to system operation on an interrupt basis. Telemetry command execution, transmission error control and message routing, and transmit message handler have been partitioned into separate tasks to avoid duplication since they have multiple uses within the system.

Messages which are transmitted over the service channel are received by the telemetry channel interface module and buffered at that point if they are addressed to the station. The hardware provides an interrupt at the end of a message indicating that a complete message has been buffered and received. This message must be moved from the buffer into the processor's memory. Data structures must be created which identify the important parameters of the message (received branch identification, message length, message location, and error status), the appropriate error control procedures initiated, and control passed to the appropriate function.

These processes are performed by the receive message handler. Its primary entry point is an interrupt from the telemetry channel hardware. Routine polling functions in which no status change is involved will occur within this task along with appropriate requests to the message transmission task for additional polls. Processing time within this task consists of a fixed period which is required for data structure building, error control functions, and transfer of control; and a variable period which is related to the message length when the message is moved from the telemetry channel buffer into the processor memory. Estimated fixed processing time is on the order ot 50 instructions and variable processing time is on the order to 5 instructions per byte.

Equipment switching has been included under the telemetry command execution task since requests for equipment switching can occur from two sources. The first source is an operator generated command for switching which will generally be a telemetry message or a local command which will pass through the receive message processor. The second source is through generated commands for switching from the fault isolation and restoral task.

In both cases, a definite set of operations is required. Prior to switching, the state of the off line equipment must be tested. If the off line equipment is in a failed state, a decision is

required to proceed with switching. This is determined by information passed to the task. Only operator initiated switching requests can force an attempt to switch off line equipment with a failed status. After the switching action has occurred, status of the equipment and information which states that the switching action did or did not occur must be passed back to the task which initiated the request for telemetry command execution.

The overall complexity of the telemetry command execution task for equipment switching is not great. As such, the execution time of this task is not anticipated to be very large. Considering the access to the data acquisition hardware and settling times for equipment switching, this task should be able to perform its task of equipment switching on the order of 100 instruction times. Equipment switching could be incorporated into the tasks which initiate equipment switching, however, by including equipment switching at this point, any changes in equipment switching procedures impact only this single section.

Several other functions are required under the scope of telemetry command execution. These fall primarily under the area of operator initiated commands. Requests for equipment data, system initialization, reacquiring of stream status, modification of equipment and connectivity tables, initiating diagnostic software all fall in this area.

Along with the common characteristic of operator initiation, these functions also share the trait of relatively infrequent utilization and comparatively loose time response. As will be discussed later under software optimization, this relaxes the requirements that are placed upon these modules. This also implies that operator commands can be integrated into the system with very slight impact on the overall system performance.

The transmission error control and message routing task is responsible for the positive error detection and retransmission of the 3 error control classes of messages which exist in the system. In general, the error control class of a message can be rapidly determined by examining the control field of the message and the first byte of the information field.

Status reporting messages and local loop telemetry messages have no requirements for re-routing upon determination of loop failure. Error control for these messages is confined to a predetermined number of retransmissions before the message is abandoned and purged from the system. Global messages must be re-routed along the secondary route after determining the primary route has failed and they also have an end-to-end ACK/NAK requirement.

Error control must be positive in both the transmit side and receive sides. Receive side error control is positive in that a message has been received and the block checksum of the protocol has a very high probability of detecting any transmission errors. The transmit side receives positive feedback in the form ACK/NAK messages which are adequate as long as the station address is not modified by errors or the message is lost. To handle lost messages, a time delay is associated with each transmitted message which will cause retransmission.

Operation on a receive message is as follows: when the message has been received, the error control status is examined. If the message has been received in error, the appropriate NAK

response is set for the receive branch, the received message purged, and memory allocated to the message is returned to free storage. If the message has been received correctly, the appropriate ACK response is set for the receive branch and control passed to the appropriate task.

Transmit message error control must identify the message with a sequence number for ACK/NAK identification and establish the retry and message class for new messages. A time interval for retransmission must be established and associated with the message sequence number. If the message is lost as determined by the elapse of the time interval, the retry number must be decremented, a new sequence number assigned and the message released for transmission. If the retry number has reached zero, the message must be re-routed if appropriate or purged and memory returned to free storage.

ACK response primarily involves canceling the time interval associated with the sequence number of the message and returning the memory allocated to the message to free storage. NAK response is identical to that outlined under the lost message response.

The transmit message handler has the primary responsibility of providing the software interface between the TSC processor and the hardware of the telemetry channel interface. All messages which are generated by a particular processor or which are being routed through this station must pass through this task. In general, this task performs the final message composition where address, control field, and message content field are joined; transfers the composed message to the telemetry channel hardware; and is responsible to maintain the requirements of the protocol for the transmit function.

Since the address, control field, and message content field are operated on within separate tasks, it is desirable to maintain these fields in areas that are easily accessible to the separate tasks and combine them when all operations on the fields have been completed. This is especially true given that multiple tasks have access to and operate on the message content field.

Interaction with the telemetry channel hardware is primarily a function of testing telemetry channel hardware for status and transferring the message to the hardware at a rate that will guarantee the transmit hardware is never left without data during a transmit period. To meet this requirement, it is necessary to completely disable the interrupt system for the brief time necessary for this transfer.

The major protocol requirement that the transmit message handler must maintain is the number of unacknowledged messages that are in the system. For SDLC, this is a maximum of 8 frames. The comparisons needed to maintain this requirement are simple and should not greatly impact the execution time of this task.

TSC system response time is a function of the telemetry channel resource, the processing throughput of the TSC processor, and the organization of the TSC hardware. The telemetry channel is a fixed resource and the hardware should not limit the system. The major area for response time thus becomes the software. Since software optimization can increase the cost of the software, it is important to carefully weigh the potential benefits of optimization of software areas in the context of the whole system operation. Some initial candidates for optimization can be identified readily. These include frequently executed sections and all sections which are potential bottlenecks. Other areas can be identified from algorithm operation.

The first major area for optimization is in telemetry channel access. Ignoring propagation times, the access time to the telemetry channel is a function of the polling rate. From the link-level protocol, only one go-ahead access can be active in the local loop at one time. This implies that the primary station releases a poll with a go-ahead and cannot release another until all the traffic with the poll has been processed.

At this moment, it becomes important to define a normal operating mode. First, the error rate specifications of the network are very good such that the large majority of the traffic is received error free. Second, the reliability of the DRAMA equipment is high and equipment failures or changes in equipment state are the occasional exception, rather than the rule. This yields a normal operating mode of poll-no change response. The software sections that are exercised are the receive message handler, the ACK section of the transmission error control task, the transmit section of the error control task, the memory management task for message buffering, and the transmit message handler.

A primary abnormal operating mode is the execution of the fault isolation and restoral algorithm. Of the three major sections of this algorithm, the status message processor and status correlation function are the driving factors. Time spent in correlation of status and routing of status reporting messages adds to their propagation time and delay times within the algorithm. Optimization within the equipment switching function would only increase performance slightly since the majority of the time spent there is in delay, waiting for equipment synchronization and status reporting messages.

The remaining software optimization confines itself to the overall area of processor utilization. In general, the lower the overall processor utilization, the lower the memory requirements for buffer area. This is because activities can be completed and removed without requiring queueing. Tasks with infrequent execution contribute very little to the overall utilization and thus require little optimization for throughput. The only consideration that must be given to these low utilization tasks is that they be confined to some reasonable length or be interruptable. The tasks not discussed with reference to throughput optimization which may require such optimization because of processor utilization are the determination of stream status within the data acquisition task and the timing function.

The recommended processor is the Texas Instruments TSM 9900. From benchmark information, this processor is substantially more efficient with respect to software than the Intel 8080. An executive program is available for the 8080 which requires 2K bytes. Similar functions could probably be performed with slightly over 1K bytes with the 9900. Given that the executive is highly optimized for speed at the expense of memory, the deployed executive for the TSC will likely be 2K bytes also. It is likely that each of the tasks at a TCU will occupy roughly 1K bytes each. Diagnostic software will probably require 2K bytes.

This implies a total software requirement of about 13K bytes including resident diagnostics. In addition, it is estimated that the CDU software (including software maintenance aids) will total about 13K bytes also.

AD NUMBER	************************	E100012 ************************SC
FIELD 2:	FLD/GRP(S)	17021
FIELD 3:	ENTRY CLASS	U
FIELD 4:	NTIS PRICES	HC MF
FIELD 5:	SOURCE NAME	E-SYSTEMS INC ST PETERSBURG FLA ECI DI
FIELD 6:	UNCLASS. TITLE	TANSMISSION SUBSYSTEM CONTROL ANALYSIS
FIELD 7:	CLASS. TITLE	
FIELD 8:	TITLE CLASS.	U
FIELD 9:	DESCRIPTIVE NOTE	FINAL REPT.,
FIELD 10:	PERSONAL AUTHORS	SMITH,RICHARD K. ;BEAUCHAMP ,JERE N. ;
FIELD 11:	REPORT DATE	15 JUL 77
FIELD 12:	PAGINATION	264P
FIELD 13:	SOURCE ACRONYM	
FIELD 14:	REPORT NUMBER	
FIELD 15:	CONTRACT NUMBER	DCA100-76-C-0036
FIELD 16:	PROJECT NUMBER	
FIELD 17:	TASK NUMBER	
FIELD 18:	MONITOR SOURCE	
FIELD 19:	MONITOR SERIES	
FIELD 20:	REPORT CLASS	U
FIELD 21:	SUPPLEMENTARY NOTE	
FIELD 22:	ALPHA LIMITATIONS	DISTRIBUTION OF THIS DOCUMENT IS CONTR ENGINEERING CENTER, ATTN: CODE R100, R NOT AVAILABLE FROM DDC. CATALOGING SUF
FIELD 23:	DESCRIPTORS	*COMMUNICATIONS NETWORKS, *TRANSMITTIN DATA ACQUISITION, FAULTS, ISOLATION, T
FIELD 24:	DESCRIPTOR CLASS.	U
FIELD 25:	IDENTIFIERS	*DCS (DEFENSE COMMUNICATIONS SYSTEM), DIGITAL EUROPEAN BACKBONE NETWORK, EUR
FIELD 26:	IDENTIFIER CLASS.	U
FIELD 27:	ABSTRACT	THIS REPORT REPRESENTS THE RESULTS OF DEVELOPING A CONCEPT FOR TRANSMISSION BACKBONE (DEB) NETWORK. INCLUDED IN TH OPERATIONS CONCEPT, TELEMETRY, DATA AC
FIELD 28:	ABSTRACT CLASS.	U
FIELD 29:	INITIAL INVENTORY	0
FIELD 30:	ANNOTATION	
FIELD 31:	SPECIAL INDICATOR	@
FIELD 32:	REGRADING CATEGORY	
FIELD 33:	LIMITATION CODES	1 21
FIELD 34:	SOURCE SERIAL	F
FIELD 35:	SOURCE CODE	393221
FIELD 36:	DOCUMENT LOCATION	7
FIELD 37:	CLASSIFIED BY	
FIELD 38:	DECLASSIFIED ON	
FIELD 39:	DOWNGRADED TO CONF.	
FIELD 40:	GEOPOLITICAL CODE	1206
FIELD 41:	SOURCE TYPE CODE	4
FIELD 42:	TAB ISSUE NUMBER	@@--@@

0012 ••••••••••••••••••••••••••SCN-- AJJ47L

MF
TEMS INC ST PETERSBURG FLA ECI DIV•
ISSION SUBSYSTEM CONTROL ANALYSIS AND DEVELOPMENT•

REPT.,
N,RICHARD K. ;BEAUCHAMP ,JERE N. ;
JL 77
P

D0-76-C-0036

RIBUTION OF THIS DOCUMENT IS CONTROLLED BY DEFENSE COMMUNICATIONS
NEERING CENTER, ATTN: CODE R100, RESTON, VA 22090. THIS DOCUMENT IS
AVAILABLE FROM DDC. CATALOGING SUPPLIED BY DCA.
MUNICATIONS NETWORKS, •TRANSMITTING, •DATA TRANSMISSION SYSTEMS, CONTROL,
ACQUISITION, FAULTS, ISOLATION, TELEMETRY

(DEFENSE COMMUNICATIONS SYSTEM), •DEFENSE COMMUNICATIONS SYSTEM,
TAL EUROPEAN BACKBONE NETWORK, EUROPE

REPORT REPRESENTS THE RESULTS OF A STUDY AIMED AT ANALYZING THE NEEDS AND
LOPING A CONCEPT FOR TRANSMISSION CONTROL FOR THE DCS DIGITAL EUROPEAN
BONE (DEB) NETWORK. INCLUDED IN THIS REPORT ARE CONSIDERATIONS REGARDING
ATIONS CONCEPT, TELEMETRY, DATA ACQUISITIION, PROCESSING, CONTROL AND REPORTING.

0

21

21

2

[illegible]